SPIRES
(Stanford Physics Information REtrieval System)
1969-70 Annual Report

to

the National Science Foundation
(Office of Science Information Service)

Project Nos.
GN 600
GN 742
GN 830

Edwin B. Parker
Principal Investigator

Institute for Communication Research
Stanford University

June 1970

TABLE OF CONTENTS

Section I

INTRODUCTION AND BACKGROUND

This is the third "annual" report to the National Science Foundation on Project SPIRES (Stanford Physics Information REtrieval system). It covers the 18-month period from January 1, 1969 to June 30, 1970. Detailed material on work completed by the project during 1967 and 1968 is contained in the two previous annual reports. For those who are not familiar with SPIRES, a background summary might be helpful.

1969 marked the completion of the third year of research and development activity under National Science Foundation grants: Gn 600, Gn 742 and Gn 830. SPIRES is funded to develop and study an online physics information system. The site of this research and development activity is Stanford University, and its users include specialized research groups and the Stanford University Libraries. Because of plans for expansion beyond physics, the P in SPIRES has been informally changed from physics to public; the Stanford Public Information REtrieval System.

Essentially the SPIRES Project is developing an augmented bibliographic retrieval capability that will initially be available to the faculty, students and staff of a major university. The traditional bibliographic retrieval system is the library. For a variety of reasons, libraries are turning to computers as a means of augmenting service to their patrons and improving their own internal processing operations. With funds from the Office of Education, Stanford University Libraries has been conducting a research and development project in the application of online computers to library bibliographic operations. Project BALLOTS (Bibliographic Automation of Large Library Operations using Time Sharing) began in mid-1967. In the interest of developing an integrated, computer-based campus information system, SPIRES and BALLOTS have been collaborating since 1968. This collaboration is formalized through a single executive committee chaired by Professor William F. Miller, Vice-President for Research and Associate Provost for Computing. Other members of the committee include: David C. Weber, Director of Libraries; Paul Armer, Director of the Stanford Computation Center; Professor Edwin B. Parker, Associate Professor of Communication and Principal Investigator for Project SPIRES; and Allen Veaner, Assistant Director for Bibliographic Operations of the Stanford University Libraries and Principal Investigator for Project BALLOTS.

SPIRES has two overall and long-range goals. The first is to provide a computer-based, bibliographic retrieval system for a variety of user groups in the Stanford community. The second goal is to support the University Libraries automation efforts by contributing to common software development. An immediate short range goal is to provide an online bibliographic information service for Stanford physicists, particularly high energy physicists. These goals must be achieved within a framework of effective, efficient operation. Effectiveness is ensured by careful study of and constant interaction with users and the user environment. Efficiency is assured by evaluation of costs and performance factors under operational conditions.

The current SPIRES system uses the Campus Facility, one of several major computer installations of the Starford Computation Center. Charts of the Campus Facility hardware and software configuration are contained in Appendix A. The system uses the following IBM equipment: a 360 Model 67 computer, with approximately one million bytes of core storage; a 2314 disk drive for storage of machine readable files; and 2741 typewriter terminals for input of bibliographic data. There are over two hundred terminals on the Stanford campus, and a locally developed time-sharing system permits approximately sixty of these terminals to be online at any one time. The Campus facility serves as the "work horse" computer in meeting the teaching and research needs of Stanford faculty and students.

Section II

THE SPIRES I PROTOTYPE

In 1967 a small one-terminal demonstration system was mounted on the 360 model 75 computer at the Stanford Linear Accelerator Center (since replaced by a 360/91) using an IBM 2250 display terminal. Following this pilot demonstration system, most of 1968 was spent in creating the software necessary for a multiple-user online prototype. This included the development of an online supervisor program (See 1968 SPIRES Annual Report), and search, retrieval and update programs. By January of 1969 the prototype version had been tested and was ready for service. In late February the prototype began on a scheduled hour and a half per day, five day a week basis. This scheduled service continued through the summer of 1969. IBM 2741 typewriter terminals were placed in the Stanford University Libraries and in the Stanford Linear Accelerator Center Library. The SPIRES system, however, can be used from any terminal on campus. At present, SPIRES is not in scheduled service, but can be loaded on demand from any terminal connected by leased line or dial telephone to the Stanford 360/67. The most frequent use is made of SPIRES by the SLAC library staff, conducting searches for SLAC physicists.

Several local files were created and searched using the SPIRES system. At the Stanford Linear Accelerator Center (SLAC) Library a file of preprints in high energy physics was created. Records of new preprints are added weekly, and a note is made of any preprint that is published. Input is via the IBM 2741 typewriter terminal in the SLAC Library. The preprint file contains approximately 6500 documents, including all the high energy physics preprints received in the SLAC Library for a period from March 1968 to the present. Input and update is done by regular library staff at SLAC. Searching is possible by author, title, date and citation. Appendix M contains a guide to searching the preprint file which describes a typical search and various system features.

Appendix B by Louise Addis records the SLAC experience in working with SPIRES. A major outcome of SPIRES use by librarians and physicists at SLAC is a clear definition of the direction which further development should take in providing optimum service to high energy physicists. The evaluation and recommendations are contained in the SLAC appendix.

In addition to its use for online search and retrieval, the preprint data base has two significant by-products. The first is an annual cumulative list of publications by SLAC staff physicists. The second is a weekly list of preprints, "Preprints in Particles and Fields" (PPF). PPF began publication in January 1969; a master copy for the list is produced each Thursday from the week's SPIRES input data set. An added feature of PPF is the "Anti-Preprint" list which records when and where previously announced preprints are published. Samples of these publications are in Appendices C and D. By early 1970 PPF was distributed to about 1600 high energy physicists and libraries in the U.S. and other countries. Costs were borne by the Division of Particles and Fields of the American Physical Society using funds obtained for the purpose from the AEC. As of July 1, 1970, PPF will be distributed only to subscribers who pay a $10.00 per year subscription fee. As of mid-June 1970, more than 600 subscriptions had been ordered, making the publication financially self-supporting.

A separate functional unit for data input and control was established in the Main Library of the Stanford University Libraries. Library staff members were trained in the use of the SPIRES system, and although five one-hour training sessions were originally planned, staff members were able to use the system effectively after only two sessions. A data base consisting of about 30% of the Main Library's monograph orders was created and updated; this is known as the In-Process File (IPF). The Library maintained a weekly input and update operation until fall of 1969. Forms and procedures were developed and statistics were collected on the use of the system. The IPF may be searched by author, corporate author, conference author, title, date and record ID number. Appendix M contains a search guide and sample search material prepared for the library staff.

In addition to the preprint and library in-process file, several other files were created. One personal file of over 500 documents was created by Professor John Harbaugh of the Geology department. A small file of documents in African history was also created. A collection of educational research documents from the Stanford ERIC Clearinghouse for Educational Media and Technology was added to the system; this file was searched locally and used for demonstrations in other parts of the country.

After several months of operational experience, the last quarter of 1969 was spent in evaluating the SPIRES I prototype system. This evaluation was conducted by members of the SPIRES and BALLOTS staff with the assistance of an independent computer consultant, Robert L. Patrick. The evaluation indicated that a major milestone had been reached with the successful operation of SPIRES I. Technical feasibility was clearly demonstrated. The special target audience of high energy physicists found the SPIRES system significantly valuable. Another user group (the Library staff), with almost no knowledge of computers was able to use the system after only a short training period. A variety of data bases were created and successfully searched from various points on campus. It became apparent that the data bases used by the SPIRES system, particularly library files and special subject files such as the preprint file, are characterized by continued growth and intensive update activity. If the SPIRES system were used on a full time basis, users would depend heavily on software and hardware reliability.

Requirements of cost and file integrity become critical in the design of a large file bibliographic retrieval system for daily use. A careful evaluation of the prototype operation, including cost and timing studies, revealed that the anticipated benefits of a production version of SPIRES (SPIRES II) could not be realized by implementing it on Stanford's 360/67 configuration. Two major problems intervene. The first is file integrity. The system software of the 360/67 as implemented at Stanford emphasizes maximum throughput at the expense of absolute file integrity. Stanford's system must handle large quantities of relatively small student jobs, and responsibility for daily file backup is placed on the user. This is appropriate for the research and teaching uses of the Campus Facility, but it is inappropriate for large production files with high update activity so characteristic of information retrieval and library automation applications. Large, update-intensive files must emphasize maximum file integrity and minimum mean time to recovery. This means that backup protection for files and update activity must be part of the system software, and recovery procedures must be fast (measured in minutes) and thorough (return to pre-breakdown condition). Loss of a bibliographic file because of a computer malfunction is not tolerable, nor is it acceptable for a library system to be without access to its essential files during a major part of a working day while awaiting recovery from some malfunction. Researchers who have created special files must have the confidence that their files will not be accidentally destroyed or inaccessible for long periods of time.

The second major problem on the IBM 360/67 is the cost of day to day operation. Analysis of the most economical option on the 360/67, adding bulk core for a dedicated partition indicated that retrieval costs would be on the order of $15 to $20 per terminal hour, depending on the estimated level of use. This is largely a function of the billing algorithm the Campus Facility requires to maintain a standard rate structure that generates sufficient total facility revenue. Heavy use of high priority cycle time during peak hours is priced much higher than overnight rates. Both the resulting unit costs and the total cost for the daily use of the system were considered too high to permit a firm commitment from the University to continue operation of a campus bibliographic retrieval system on a production basis.

Section III

DEVELOPING A PRODUCTION SYSTEM--SPIRES II

In meeting the daily information retrieval requirements of a large user group including the library, the SPIRES system must operate in a production environment. In an online production environment, a variety of research users access a variety of data bases throughout the day. In addition, from ten to twenty professional terminal input personnel use the system for all or part of the day. This places heavy requirements on the system for absolute reliability, file integrity, and rapid recovery as well as cost acceptability.

A sophisticated search and retrieval capability, which is not available a good share of the time, is of little use either to a faculty member or a librarian. Absolute reliability means that the system must have a minimum amount of "down time." Periodic machine or program breakdowns, which idle ten to twenty terminal input personnel and make private data files inaccessible to ongoing research projects, are not acceptable. However, in the event of system malfunction, existing files must be protected and recently input data must be preserved. Rekeyboarding of data destroyed in a system failure adds prohibitive costs and time for file restoration. Hence, the necessity for file integrity and rapid recovery. A computer system must be able to handle increasing processing loads in a manner which is cost acceptable to a variety of users. Delays or inability to gain access to library computer files due to heavy CPU use is not acceptable in a bibliographic oriented file system. Developing a production oriented retrieval system requires a comprehensive and formally defined system development process. Appendix E describes in detail the system development process for SPIRES II.

This system development process has six overlapping phases, each with a discrete and specified output. A graphic representation of the activities in each phase of the process is contained in Appendix F. The overlapping relationship of the development phases is shown in the System Products Chart in Appendix G. The six phases of the system development process are:

A. Preliminary analysis
B. Detailed analysis
C. General design
D. Detailed design
E. Implementation
F. Installation.

Preliminary Analysis is undertaken as a basis for detailed enumeration of the system's requirements. During this phase, policies in support of development are established, goals are defined and the user environment is characterized. The current system is documented and analyzed for its limitations including cost factors. A long-range scope is stated which deals with these limitations. A sub-scope for first implementation is selected by choosing a combination of elements which yields an optimum use of resources by matching areas of critical need with available funds. The results of the preliminary analysis phase are contained in a formal Scope Document. The SPIRES and the common software sections of the Scope Document are reported in Appendix J and Appendix L, respectively.

The Detailed Analysis Phase enumerates the complete functional requirements of the production system. Performance requirements are stated quantitatively including such factors as response time, hours per day of online accessibility and maximum allowable down time. Record input and output are estimated in terms of volume, growth and fluctuations. All input/output documents are laid out in character by character detail. Processing rules which transform input data elements into output data elements are specified, and cost limits are established. The results of the Detailed Analysis Phase are presented in a requirements document.

There is considerable phase overlap and some minor requirement tasks carry over into the next phase and some design tasks are begun in the Detailed Analysis Phase.

In the General Design Phase alternative software-hardware configurations are conceptualized to meet the requirements of the SPIRES II system. Each configuration is analyzed and the alternative which yields the optimum combination of advantages is selected and detailed in a General Design document.

During the Detailed Design Phase the development staff executes a detailed internal design down to the program module level. The General Design document is a basis for this work. A complete set of programming specifications is produced as well as a test plan, a training plan and an implementation plan. These are presented in a Detailed Design document.

During the Implementation Phase programs are coded according to specifications and tested. Training courses and materials for both the manual and automated portions of the system are completed. The users go through a period of intensive training.

As the Installation Phase begins, files are converted. Where manual procedures are being replaced, there is a short period of parallel operation and then a complete cutover is made to the new system. Performance statistics are gathered over a ninety-day period in the production environment. All final documents necessary for research reporting and continuous operation of the system are prepared.

All activities which occur during the system development process are scheduled and evaluated at key milestone points. A graphic presentation of these is given in the major milestones and schedules chart in Appendix H. Internal target dates may be modified but the date of production operation is fixed.

In addition to an overall development plan, an appropriate project structure and management are necessary for the creation of a production system. In November 1969, A. H. Epstein was appointed director for both the SPIRES and the BALLOTS system development activities. Mr. Epstein holds a joint appointment with the Stanford Computation Center and the Stanford University Libraries. As Project Director, he reports directly to Paul Armer, Director of the Stanford Computation Center, and he is also Chief of the Stanford University Libraries Automation Department. Mr. Epstein came to Stanford from private industry where he held a senior management position involving the development and operation of online information systems. His previous work included the application of innovative technology, such as Computer Output Microforms (COM).

The staff of both SPIRES and BALLOTS is consolidated and under the direction of Mr. Epstein. In January 1970 the staff was relocated in new quarters adjacent to the Computation Center. Phase A tasks in the system development process began in late 1969 and continued into the first quarter of 1970. A formal project management system is in operation. Tasks are defined, assigned and coordinated using task control sheets, schedules and full documentation requirements. A separate documentation unit has been established within the project to assure that reporting requirements are met and that the system is fully documented for maximum research and operational value. The Organization Chart in Appendix I shows the project structure and staff.

Section IV

SYSTEM SCOPE AND REQUIREMENTS

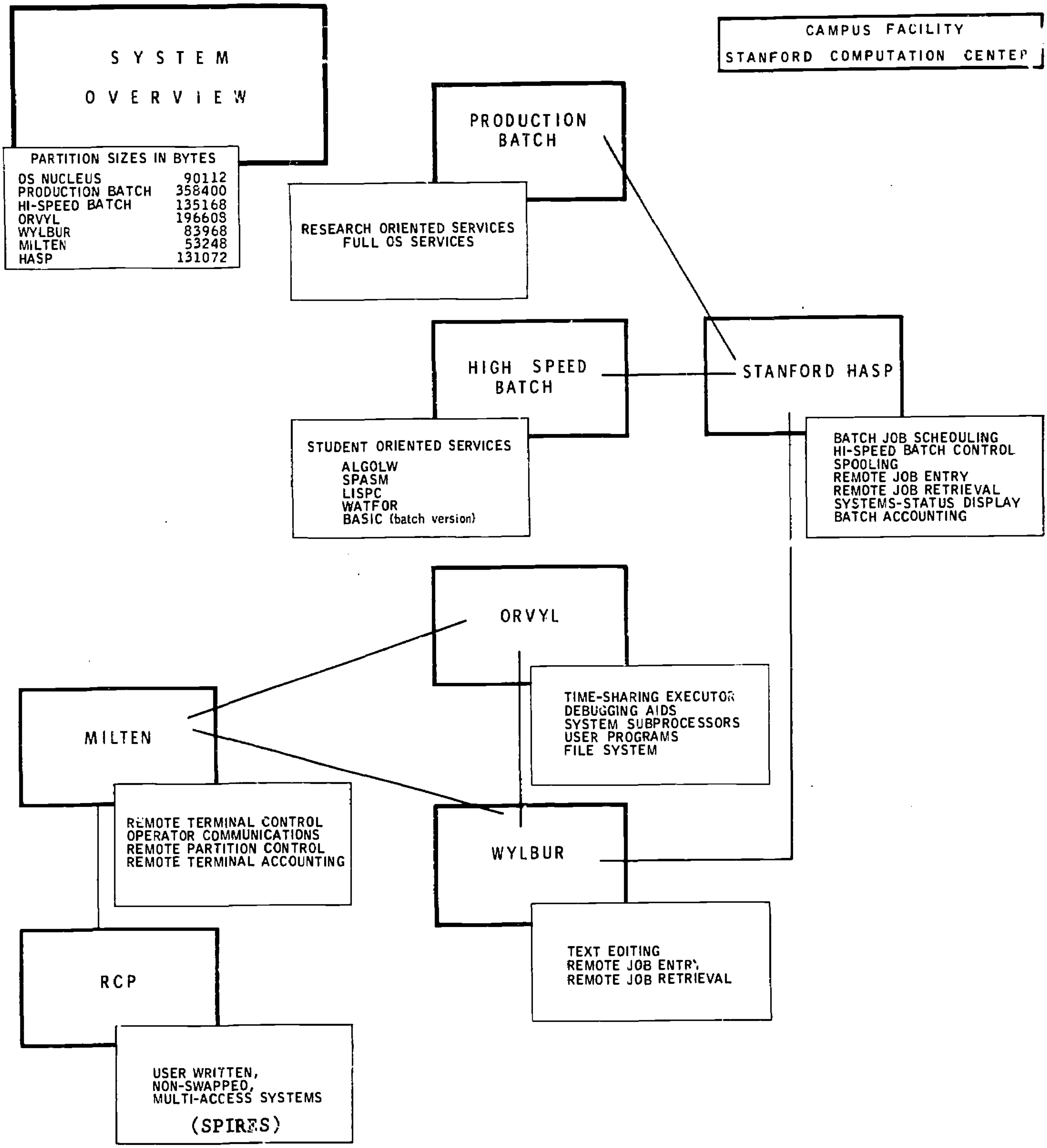
The first phase of the SPIRES II system development process was completed during the first quarter of 1970. This phase was documented in a 160 page document entitled "System Scope for Library Automation and Generalized Information Storage and Retrieval at Stanford University." (Copies are available from the project at $7.50 prepaid.) Part 3 of the Scope Document, which discusses the SPIRES II generalized information storage and retrieval system, is attached as Appendix J. The Scope Document characterizes the users and the user environment and summarizes the limitations of SPIRES I. It describes a long-range scope of retrieval and file management capabilities as well as a first implementation scope (SPIRES II). A unique feature of the Scope Document is a tutorial appendix, which describes the information storage and retrieval concepts underlying the SPIRES/BALLOTS system. This is attached as Appendix K.

A key element of the Stanford system is the concept of shared facilities. These are the common software-hardware facilities which will service SPIRES II, BALLOTS II and eventually, additional applications. Examples of shared facilities are an online executive program and a text editor. Shared facilities are discussed in the Scope Document and the section describing them is attached as Appendix L. The operations environment in which the common software will serve various applications is called the Data Facility. The computer configurations selected for the Data Facility will be large enough to service SPIRES II and BALLOTS II efficiently and effectively with potential for later growth as required.

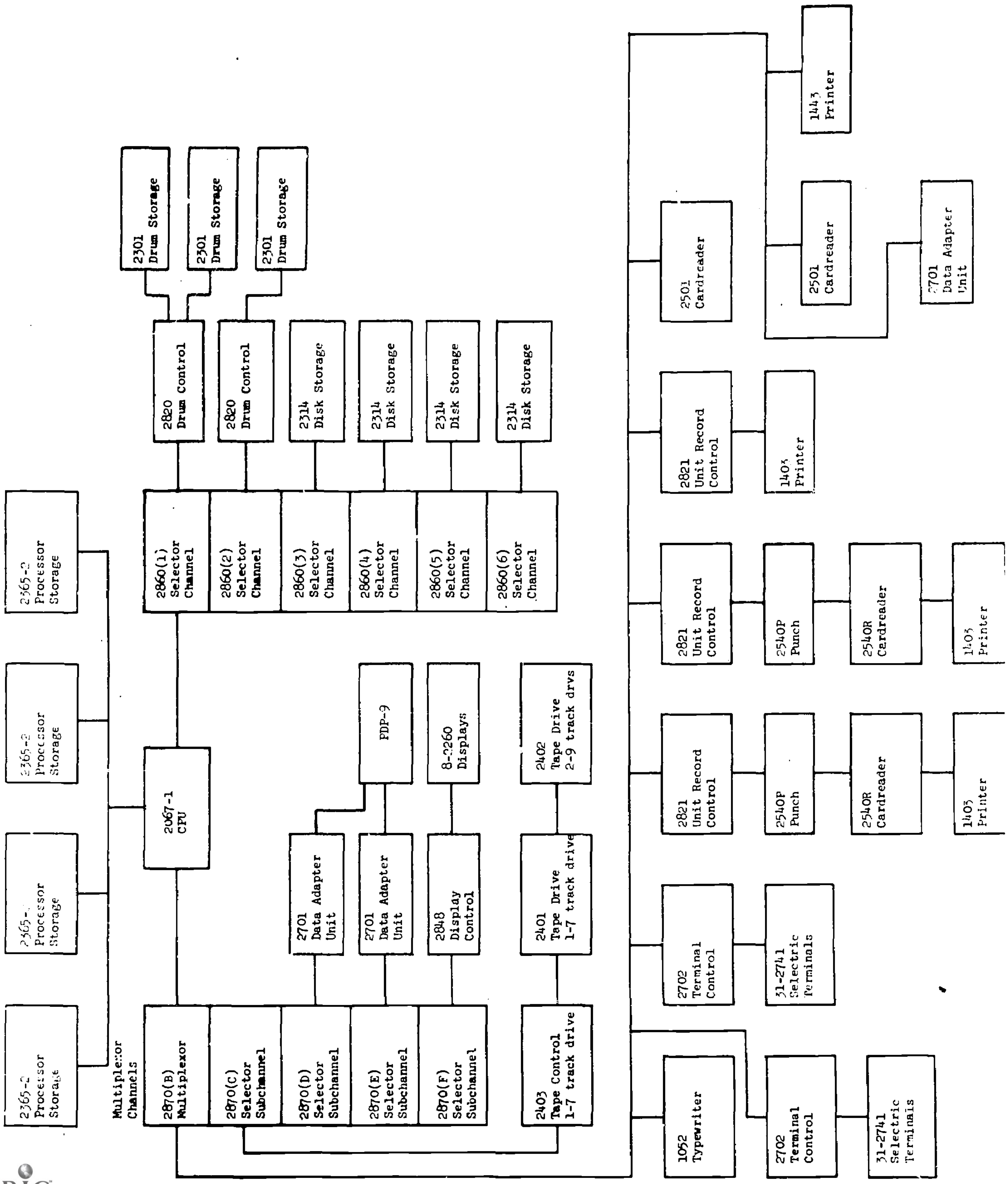
As this report goes to press, the project is deeply involved in the Detailed Analysis Phase. This is a crucial phase in system development because system requirements (such as performance and output documents) are established and approved by the project and system users. Requirements analysis involves almost daily contact with identifiable user groups, in this case librarians, and painstaking review of details to assure compatibility with the users' operational needs. For example, each separate visual display format is designed jointly by a team of librarians and analysts. Each week library department heads and key supervisory personnel meet with the Project Director and various staff members. Topics range from discussion and approval of written statements of system assumptions (such as hours of online operation and types of file access) to discussions of system flowcharts.

A variety of technical tasks are being carried out. Existing programming languages and online software are being evaluated. The system is being simulated to determine, for example, variation in response time under various processing loads. An online command language is being written for search, retrieval and update. An analyzer is being designed to parse the language and produce appropriate diagnostics. A requirements document incorporating elements of design will be produced later this year. It will be formally approved by the Project Director, the principal investigators, and the users. This requirements document will be the basis for detailed system design and programming.

APPENDIX A: Stanford Campus Facility Software and Hardware



Campus Facility 360/67 Machine Configuration



APPENDIX B: Stanford Linear Accelerator Center Participation in Spires by Louise Addis

STANFORD LINEAR ACCELERATOR CENTER PARTICIPATION IN SPIRES

The special characteristics of SPIRES as a Physics Information Retrieval Project were outlined by E.B. Parker in the 1967 SPIRES ANNUAL REPORT, as follows:

> "Five features characterize the SPIRES project and serve to distinguish it from other on-line information retrieval projects. The first is the strong behavioral science emphasis . . .
>
> The second distinguishing feature is the data base to be used in the system. The first criterion for selecting the data base is to be responsive to user needs, finding out user priorities rather than starting with assumptions that may not apply locally. . .the second criterion ... is to take advantage of whatever data bases are available in machine readable form that may be of some value to our users. . .
>
> The third distinguishing feature of the SPIRES is its focus on the development of adequate computer systems software and applications programming. . .
>
> The fourth distinguishing feature can be stated negatively. There is no local manual indexing. It is felt that what manual indexing is done would, in the interests of standardization, be better left to the developing national systems rather than attempting to index at a local level. Instead the concern is with adapting to on-line retrieval whatever indexing procedures are available or can be made available, and with indexing that can be done by computer (e.g., using title words in conjunction with word stemming and synonym dictionary procedures and using citation indexing procedures)...
>
> The fifth distinguishing feature is the nature of the liaison with relevant library operations and library automation projects. The project has excellent liaison with the SLAC Library . . ."

In keeping with the basic philosophy of SPIRES, the needs and priorities of potential SLAC users were explored in a series of interviews with SLAC physicists. A summary of their response is found in the first SPIRES ANNUAL REPORT.

In accord with interview findings, high priorities were given to the following data bases:

1. SLAC preprint collection
2. Nuclear Science Abstracts
3. Journals (at that time it was thought that the T I P tapes would be available to SPIRES)
4. DESY High Energy Physics Index

The DESY INDEX was later moved up into second place as the excellence of its keyword indexing and the completeness of its coverage of the high-energy physics literature became evident. A sample data base of NSA was created but a full NSA data base was moved down on the priority list because of its size. NSA, with its interdisciplinary coverage, contains on the order of 50,000 entries/year (against the 9,000 entries/year of the specialized DESY INDEX). Journal tapes have not yet been available at a reasonable cost; however, the high-energy physics journals are thoroughly covered in the DESY TAPES.

We believed then and still do that the SLAC PREPRINT COLLECTION plus the DESY INDEX would most closely meet the goals of providing a specialized user population (SLAC and Stanford high-energy physicists) with access to:

1. The most timely information -- preprints.
2. A large enough specialized data base to permit exhaustive retrospective searches.

The choice of these two high-energy-physics data bases would allow comparison of the effectiveness of two types of subject search:

1. Title word, author, and citation searching in a file (preprints) in which no manual indexing had been done.
2. Keyword, title word, and author searching in a file (DESY) in which extensive professional keyword indexing was provided.

Citation search capability for the preprint data base was regarded as particularly important since no "manual" indexing was planned for that file. The presence of citations would allow another subject approach (in addition to title word) to preprints. Libraries on the Stanford Campus were already subscribing to the vast, interdisciplinary SCIENCE CITATION INDEX in its printed version (3,000,000 citations/year, approximately $1200/year).

In physics, the citation search has several utilities:

1. General subject searching.
2. Tracing the fate of a specific piece of work.
3. Checking on whether a particular author is doing work that others find useful. ("Publish or perish" is giving way to "be cited or be sunk".)

As more physicists discovered SCI's purpose and utility, we found ourselves struggling through more and more manual searches in its profoundly unsatisfactory pages (the print is submicroscopic, it is always far behind, and references are skeletal and must always be looked up again in a second source to locate titles). We welcomed the potential capacity of SPIRES to allow us easily to bring these citation searches up-to-date in our own preprint collection (a year or more ahead of the printed index).

Originally it had been planned to allow citation searching in the same detail as in the printed SCI (by author and all types of papers). This proved technically difficult and the input too time consuming. We therefore limited citation input to bona fide journal references which could be entered and searched as simply a CODEN (for journal title), a volume No., and a first page No. Since the references on preprints are frequently sloppy and inaccurate, and since they will eventually appear in the printed SCI, this compromise seems a reasonable one to make. It does, however, make it impossible to do citation searching on conference papers and on preprints, and, of course, we cannot do a citation search by author.

The ultimate SPIRES system, should, of course, allow for the inclusion of the complete SCIENCE CITATION INDEX... if only for the benefit of the Medical School where it is perhaps most heavily used in printed form.

Then, to reiterate, our goal as SLAC users was creation of data bases of the most timely material, and one large enough and complete enough (with a professional subject index) to allow a thorough search to be made on any high-energy-physics topic. The chosen materials were:

1. SLAC preprint collection (3,000 documents/year)
   Searches to be utilized:
   a. Author
   b. Title word
   c. Report number
   d. Citation
   e. Date

2. DESY HIGH ENERGY PHYSICS INDEX (9,000 documents/year)
   Searches to be utilized:
   a. Keyword (up to 23 assigned to each document)

b. Title word
c. Author
d. Date

Since March 1968 a data base containing the SLAC Preprint accessions has been regularly created and maintained (weekly as permitted by hardware and software development). Input has been via the 2741 terminal located in the SLAC Library.

This preprint data base currently contains bibliographic information and citations for some 6500 documents, including all the high-energy physics preprints received in the SLAC Library for the period March 1968 to the present. Approximately 1000 documents are reports, preprints, and translations produced by members of the SLAC staff. The annual cumulative list of SLAC publications is produced from the SPIRES data base by a batch program.

Specifications for the conversion of the DESY FILE to the SPIRES format were completed in June 1969 (see list of SLAC-SPIRES documents). Though the programming has been nearly completed for the conversion, the data base has not yet been created.

In late 1968, SLAC proposed to and received a special grant from the AEC to begin printing and mailing (under the sponsorship of the Division of Particles and Fields of the American Physical Society) a weekly list of preprints "Preprints in Particles and Fields (PPF)". PPF began publication in January 1969. Master copy for the list is produced each Thursday from the week's SPIRES input data set.

PPF is currently used by nearly 1600 high-energy physicists and preprint libraries in the Western Hemisphere (including SLAC). The results of a questionaire sent to subscribers indicates that PPF is a success among high-energy physicists. (One enthusiastic user described it as "the best thing to happen in physics information in 50 years"). A popular feature is the "Anti-preprint" list which lists when and where previously announced preprints are published. Though PPF is not an integral part of the SPIRES system but a byproduct (which we would produce anyway, though more laboriously, without SPIRES), the enthusiastic response of the wider high-energy physics user community to "even a listing" of preprints is significant.

USER EXPERIENCE -- SPRING 1969

The SPIRES search and the preprint data base were sufficiently developed by Spring 1969 to put to the test of actual physicist users. (At that time, SLAC had only 2 or 3 on-line terminals outside of the library whereas there are now 23 such terminals.)

About 1200 people are employed at SLAC. The SLAC Library has a staff of 11. The "user population" for a SPIRES (with only a high-energy physics data base) consists of some 90 Ph.D. high-energy physicists (including about 20 temporary visitors from other labs), 25 graduate students (Ph.D. candidates) and up to 8 members of the SLAC Library staff.

The two-mile linear electron accelerator itself is a scientific instrument used by experimental high-energy physicists to conduct their research. Theoretical high-energy physicists do not use the accelerator but concern themselves with explanation and prediction. Since a high-energy physics experiment on a large accelerator may cost in the $100,000 range to perform, it is essential that work not be duplicated or undertaken unnecessarily. Therefore, keeping up (with preprints) is essential to the high-energy physicist.

The physicist users are as a group:

a. Very busy, irregular in their working hours Experimentalists, for instance, must work all night sometimes. Theoreticians usually arrive around 10:00 am and frequently work at home.
b. Quick thinking and quick learning.
c. Familiar with computers and likely to have a typewriter terminal close by (there are 23 terminals now at SLAC).
d. Interested in any real help they can get in keeping abreast of the information explosion.

As a part of the campaign to attract users, Prof. E.B. Parker spoke at a seminar. Some 15 physicists asked the SLAC Library to conduct searches for them and probably another 15 experimented with the terminal search themselves (though that was hard to keep track of). Several expressed their opinions in writing to E.B. Parker (I've attached a few of these letters of which I received copies).

The results of the user experiments with SPIRES in April and May 1969 may be summarized as follows:

1. The quick search response time of SPIRES was universally admired and the slow printout on the terminal was found universally annoying.

2. The plans for CRT devices, the save, and off-line print capability were heartily endorsed. Once the search points have been determined, the user usually doesn't wish to have to wait for printout at the terminal. He'd like his secretary to printout a WYLBUR dataset or pick up some printout at the Comp Center. He'd also like to be able to "flip through" a lot of

entries as you are able to do on the CRT, and sometimes save a few entries in a file of his own.

3. Almost every search included one or more citation elements.

4. Since the preprint data base was the only one available, no comprehensive retrospective searching could be done on-line. Consequently, much supplementary manual searching (in the DESY INDEX) was done by the SLAC Library staff (resulting in a serious work overload) during this period. Users were pleased with the results and it seems obvious that were DESY available on-line and publicized, many information needs would be better met. (We don't have the staff time to offer this kind of manual search service to everyone who needs it now and physicists don't have the time to do manual searches themselves except under the most desperate circumstances).

5. The hours 8:15-9:30 a.m. were awkward ones for physicists. If only an hour or so of on-line SPIRES service were to be available, the late afternoon would be the best for physicists. Also, in many cases, an hour was not enough time to complete the listings for a particular set of searches though the searches themselves might have taken only a few minutes.
A 24-hour day, 7-days a week availability would be the most popular. An 8-hour day, 5-days a week next. A 2-hour service during the 4:00-6:00 p.m. period next.

6. Physicists would still like to be able to save selected references in their own files, and several of them would like some form of SDI.

7. Many users mentioned the desirability of left and right truncation on all indexed elements.

An INTERIM SPIRES FOR SLAC USE:

The current version of SPIRES with the following improvements would provide SLAC with a fairly versatile on-line information retrieval system with which to gain user experience during the next 18 months, and one for which a case for some funding might be made to our budget department:

1. Completion of the Anti-PPF program (1/2 done) would save 10/15 hours per month of the preprint librarian's time and nearly that much of terminal time (while adding an undetermined amount of program running time).

2. Addition of the DESY DATA BASE would allow thorough retrospective searching on high-energy physics topics. (The implementation of No.3 below is, however, necessary to allow use of the DESY FILE). It would undoubtedly save many hours of reference librarian time and allow us to provide our users with a much more efficient subject search service. The experience which could be gained from physicists actually using a large file would be helpful in planning the future SPIRES. In connection with the DESY file, we need frequency statistics for keyword usages (per my memo of 7/22/69 to Jim Marsheck).

3. The addition of an off-line print capacity would render the use of the current SPIRES system economically feasible. Frequently the listing of 75-100 documents may be required after a search which took one minute. To be paying $9 to $16/minute for a terminal listing (as opposed to further searching) is simply not economically feasible... even in the case where several terminals are being used at one time (a rather complex scheduling feat). On-line search capacity is essential for setting up a given search. Ideally the search results should be stored in a WYLBUR data set and listed from the terminal later...but given the impossibility of this, print off-line is a satisfactory substitute.

The following additional improvements would be helpful but not essential for the interim SPIRES:

1. The addition of a message of the day to be set by the SLAC data manager for the preprint and DESY files, allowing a report to the user on the latest additions to the file, or any other relevant information. At present, the user has no easy way of knowing what material may have been added to the file since his last search.

2. Clean up of the "type own" display format to eliminate the print-out of unabbreviated element names. The user, who knows enough to choose the elements he wants printed out, can get by without any identifying tags for the sake of faster print-out.

3. The availability of a batch program which uses the "Anti-preprint" data sets to add publication notes (PBN) to entries in the data base. (Space has been dummied in as an NSP element with each preprint entry.) After a preprint has been published it is much more useful to the searcher to have a journal reference than a report number (which he must check in the card catalog to locate).

4. The elimination of duplicate entries within the DESY data base (this problem is described in detail in the DESY User Spec) and perhaps the linking of entries between the DESY and preprint files.

We envision the interim SPIRES as an on-demand system...the "demands" being made to the SLAC Library where search times could be scheduled for convenience to users and economy to the system. If the PREPRINT and DESY data bases were both available, with an off-line print capacity, we would publicize the subject search, encourage physicists to submit search questions and to use the system themselves during "up-time". We would also expect to prepare a few experimental user profiles (R.E. Taylor and B. Richter would like to be guinea pigs for such a project) to see if individualized lists of new high-energy physics documents could be sucessfully prepared using the search points available in these two files. Faculty members at CALTECH have also expressed interest in an arrangement allowing them to submit searches to SPIRES from time to time, probably via the SLAC Library.

THE ULTIMATE SPIRES

We envision the long-range SPIRES as a 24-hour/day, 7-day/week service, utilizing CRT, allowing individuals to create their own files, either from scratch or by copying out of larger data base files, and allowing users access to a spectrum of large special-subject data bases. A list of machine readable reference services most of which are currently available in printed form on the Stanford campus is attached to this document. (It would be interesting to poll the other science libraries, including Medicine to see which indexes they'd most like to have on-line).

It is, of course, essential that the cost to the user of the ultimate SPIRES be "reasonable."

INFORMATION RETRIEVAL

Certainly the ultimate SPIRES should be able to accommodate the SCIENCE CITATION INDEX as well as the more conventional indexes. At SLAC, we would hope for the eventual conclusion of the following large data bases:

1. PREPRINTS
2. DESY
3. NUCLEAR SCIENCE ABSTRACTS
4. SCIENCE CITATION INDEX (Physics and technology sections)
5. U.S. GOVERNMENT RESEARCH AND DEVELOPMENT REPORTS
6. STAR (NASA)
7. PHYSICS JOURNALS (AIP)
8. CHEMICAL ABSTRACTS (some subset of)
9. ENGINEERING INDEX (if available)

The first four of these are the most important to us.

It would seem reasonable that the ideal SPIRES be designed to accommodate any and all of the available machine readable records for which there were sufficient need among Stanford users.

The ultimate SPIRES also should allow the user or the user's "agent" such as the library, to maintain "profiles" of the user's information interests. These should be easily changeable, should be in the regular SPIRES search format (i.e. a Jones, J. and not a Smith, etc.) and should be automatically activated when new material is added to the file. Formating of the output from the profile searches will be very important since it must make very clear to the user which elements in his profile are producing "hits" and which are not.

Experience gained using a relatively large file during the interim SPIRES should be utilized in the design of the SDI features of the ultimate SPIRES. It would be desirable to draw heavily on the experience of the Lawrence Radiation Laboratory group using NSA for SDI experiments.

LIBRARY ROUTINES

Eventually, we should like to be able to "check in" the preprints received, on a SPIRES terminal rather than in our manually maintained file. We wish to "weed" with the aid of SPIRES instead of entirely manually as at present. (Now the preprint librarian personally compares the Tables of Contents of each new physics journal with our preprint holdings to locate published preprints.) Ultimately, we hope that a "weed list" can be prepared weekly by SPIRES from a comparison of new journal tapes with the preprint data base. The preprint librarian can check the "weed list" for mismatches. The preprint data base could then be updated (PBN added) and master copy for an anti-ppf be produced.

To eliminate double input, we need to produce catalog cards (or a cumulative book catalog) for our preprint collection. (We prefer catalog cards at present.) Ability to produce catalog cards from SPIRES input would allow us to consider conversion of our entire cataloging operation to "SPIRES". Conversion of our manual circulation system to an on-line (or batch) scheme might logically follow. (Currently, circulation files are maintained by call number and by borrowers names.)

EDP methods have been used for serials handling in the SLAC Library since 1963. At present all but two staff members participate more or less regularly in projects involving either keypunching or on-line data set creation. On the whole, attitudes are favorable toward further ventures into automation.

A POSSIBLE INDIRECT SLAC SUBSIDY TO SPIRES

The thorough exploration of the possibility of our using our own time-sharing system (CRBE) to create weekly preprint data sets which could then be transferred to the campus facility for incorporation into the SPIRES data base. I have explored this possibility enough to find that it is a good deal less convenient than our current system and might run aground on some technical difficulties. (data set size limits) Discussions are needed between a member of the SPIRES programming staff and the SLAC Computation Center, however, to determine whether it could indeed be done and how much programming would be needed to make it possible.

Moving the SLAC-SPIRES dataset creation to the SLAC computer would allow us to provide a large indirect subsidy to the SPIRES project without actual transfer of funds.

SLAC-SPIRES DOCUMENTS -- Formal and informal

A. INPUT FORMAT

1. Computer Note No. 30, INPUT FORMAT FOR SLAC PREPRINTS, LA, 28 Nov 1967.

An annotated version of this note is kept current (by hand) in the SLAC Library. (It needs to be reissued in a formal revision.)

2. COMMONLY USED CODEN

3. Title symbol conversion list and hyphenation conventions for physics preprints.

4. Brief Outline Guide to Wylbur for operator reference.

B. PREPRINTS IN PARTICLES AND FIELDS, a weekly newsletter in two parts

1. PPF (the preprint announcement section)

a. PREPRINTS IN PARTICLES AND FIELDS FORMAT SPECIFICATION, LA, Dec 1968.
(Program was written by Ken Siberz, Jan 69, which creates master copy for PPF according to specification)

b. PROCEDURES FOR USING PPF LIST CREATING PROGRAMS, LA, current.

c. Time and length job records.

2. ANTI-PPF (the section announcing publication of ex-preprints)

a. SPECIFICATIONS FOR 'ANTI-PPF' LIST PRODUCING PROGRAM, LA, Oct 69.
(Programming is not yet finished for this application.)

C. UPDATE

1. CURRENT PROCEDURES FOR UPDATING THE PREPRINT DATABASE USING SLAC INPUT DATA SETS AND THE SPIRES PROGRAM.

2. PROCEDURES FOR CHECKING THE BUILD AND HANDLING CORRECTIONS.

3. TIME, AND LENGTH, AND JOB RECORDS.

D. SLAC PUBLICATIONS LIST

1. USER SPEC FOR SLAC PUBLICATIONS LIST, LA, Dec 68.

The SLAC Publications lists are an annually produced cumulative listing of all preprints, reports, translations, and internal reports done at SLAC.

LIST A -- is a cumulative listing of all SLAC preprints, reports, and translations currently. This amounts to about 1000 entries in the "Preprint Data Base" by author, Report No. and by subject. Master copy for list A has been produced twice and published since the programming was completed.

LIST B -- is a cumulative listing of all the SLAC internal reports (Technical notes) by author, Report No., and keyword.

LIST B has never been produced. The input dataset containing some 600 entries has been ready at SLAC since August 1969. It has never been added to the preprint data base... initially because of technical limitations on the size of the data base and currently because of uncertainty about the immediate future of the SLAC role in SPIRES.
The TN entries are the only ones which have actually had keywords assigned locally by the SLAC Library cataloger (using the DESY KEYWORD system).
We had hoped to have a data element level update available before committing the TN's to the data base since we would like to experiment with the effectiveness of the keywords and change them at will.

E. CATALOG CARDS

1. SPECIFICATIONS FOR USING SLAC INPUT DATASETS TO PRODUCE CATALOG CARDS, LA & KB, Aug 1968.

This card-producing specification with a few minor revisions is still valid for producing catalog cards for the SLAC Library catalog. A few decisions remain to be made -- the type of card to use...whether to produce cards on the 2741 terminal or on the line printer. . . how to handle the name authority list. At the present time we are doing "double input" as a part of participation in SPIRES...one staff member continues to make catalog cards (using a stencil and a cardmaster) while the terminal operator inputs the same information into a WYLBUR data set. Programming time has never become available for this application.

F. SEARCH

1. QUICK GUIDE TO SPIRES PREPRINT SEARCH, LA, Jun 69.

ATTACHMENT TO APPENDIX F

Excerpts from letters to E.B. Parker commenting on the SPIRES system as viewed by physicists.

Letter dated 7 April 1969 from H. Saal, Experimental Group C

"I would like to take this opportunity to comment on the SPIRES system now operating at Stanford Linear Accelerator Center.

I very much appreciate this existing facility, and look forward to its expansion and growth in the future. Particularly in the field of high energy physics, where selective access to large numbers of preprint data prior to formal publication is critical, such a tool is welcomed.

Certain current limitations, such as the lack of uniform keywords, need to be overcome before the system can reach its full potential. I hope this effort will continue to be supported, and new features implemented in the manner. . .described to me."

Letter dated 21 April 1969 from D. Yount, Experimental Group D

"This note is to express our appreciation for the work you and others have done in developing the SPIRES system.

The streamer chamber group at SLAC is in the midst of a comprehensive article on meson photoproduction, and already we have used the SPIRES system to good advantage. Among the listings we have requested are: RHO Title Search (68 documents), RHO PHOTOPRODUCTION (13 documents), and articles referring to our own report, Phys. Rev. Letters 21, 841 (1968) (5 documents), which appeared some seven months ago. In each case, the lists have included the most recent and most inaccessible references, thus permitting a more thorough documentation than would otherwise be practical. We look forward to the expanded data base and increased flexibility that we understand are included in your future plans for the SPIRES system."

Letter dated 4 April 1969 from E.L. Garwin, Group leader, Physical Electronics

"I have looked at the SPIRES information retrieval system which you have been developing, and am very enthusiastic about the potential of this kind of system to aid not only my own work but the work of applied physicists generally. Applied physicists have a particularly acute need for extensive and rapid bibliographic information services and should find your kind of interactive retrieval system very helpful.

I am especially interested in the citation indexing capability demonstrated in the current SPIRES preprint data base. It is, for instance, a great time-saver for users to have titles and sources of citing articles instantly available.

SPIRES will be most useful for my own work when it has a large collection of references, for example, a five-year accumulation of "Nuclear Science Abstracts," at least a two-year accumulation of the "Science Citation Index," and ideally, several years of "Chemical Abstracts."

I hope you are able to obtain continued support for this important development effort."

Letter dated 9 May 1969 from S. Drell, Deputy Director, SLAC

"I should like to congratulate you on the contribution which the development of the SPIRES system is making to the easing of the information crisis in science, particularly in high-energy physics, here at Stanford.

The ever-growing flood of preprint and journal literature makes it essential for the physicist to have quick, direct access to the relevant literature of his field. He may then spend his time working rather than searching, confident that he is tackling something new rather than duplicating the old.

The SPIRES concept of the comprehensive on-line search with output available on a CRT-scope should provide just such a mind-augmenting system for information retrieval. Even at its present stage of operation as a prototype system only, SPIRES shows great power and flexibility and has provided what I asked of it in connection with my own research efforts.

The title work search combined with the citation search is an effective technique for exploring the high-energy physics preprint collection which has been, until SPIRES, inaccessible by subject. Several years of DESY HIGH-ENERGY PHYSICS INDEX and NSA files would, of course, greatly enhance the value of the system for searching. The inclusion of extensive SCIENCE CITATION INDEX files would benefit not only physicists, but the whole campus scientific community.

I hope that SPIRES will continue its development along the lines presently proposed. Such a system has much to contribute to easing the flow of information and ideas in all fields."

Letter dated 10 May 1969 from Prof. A.H.Rosenfeld, Secretary, Division of Particles and Fields of the American Physical Society.

"Professor Panofsky and I want to thank you on behalf of the APS Division of Particles and Fields for the major contribution made by the SPIRES project to the success of our publication "Preprints in Particles and Fields (PPF).

As you know, we recently conducted a survey of our 1500 subscribers and received an overwhelmingly favorable response to PPF. Several physicists believe PPF to be the most useful advance in physics information in the last decade.

Also, I know that Si Pasternack, the Editor of the Physical Review is enthusiastic about the PPF way of dealing with the preprint problem and himself uses "Anti-preprints" extensively in editing the references in papers for the Phys. Rev. (Journal editors have in the past been in strong opposition to other more formal preprint handling schemes.) Of course all journals have this problem of updating references to preprints. . .

I understand that additional SPIRES efforts are plannned in connection with the "Anti-preprints" section. This will help in further easing the burden on SLAC Library personnel in the production of this bulletin which is such a boon to communication among high-energy physicists."

APPENDIX C: Preprints in Particles and Fields

29/30

PREPRINTS
in Particles and Fields

P. O. BOX 4349
STANFORD, CALIFORNIA 94305

19 JUNE 1970 PPF-70-25

PREPRINTS IN PARTICLES AND FIELDS (PPF) lists new high-energy physics preprints received during the past week at the Stanford Linear Accelerator Center Library. It also provides, in the "Anti-Preprint" section, references to published versions of former preprints.

To obtain a copy of an item on this list, check your own preprint library or write directly to the author. PLEASE *DO NOT* REQUEST PREPRINTS FROM SLAC, except, of course, those by SLAC authors. "Print" and "Rx" report numbers are assigned by SLAC to unnumbered preprints and should *not* be used in requests or references.

PPF is published *weekly* by the SLAC Library in cooperation with the Division of Particles and Fields of the American Physical Society. It is sponsored by the U.S. Atomic Energy Commission Division of Technical Information. The text is produced on a time-sharing computer system through the courtesy of SPIRES (*S*tanford *P*hysics *I*nformation *RE*trieval *S*ystem) and the National Science Foundation.

High-energy physicists and preprint libraries in the Western Hemisphere may request PPF from:

Stanford Linear Accelerator Center Library
Attn: PPF
P.O. Box 4349
Stanford, California 94305

If your address is going to change soon, please fill in your new address below and return this *whole* sheet to us!

PLEASE CHANGE MY ADDRESS TO: ______________________

____

____

Preprints in Particles and Fields

Report No.	*	Title	Authors
Print-70-1413 (ALBERTA)	T	GILBERT-LIKE DISPERSION RELATIONS FOR THE FORM-FACTORS. n.d. 8p.	A.N. Kamal (Alberta U.)
ANL/HEP-7008	ET	LECTURES ON WEAK INTERACTIONS. Apr 1970. 26p.	R.J. Oakes (Northwestern U.)
BNL-14703	T	DENSITY MATRIX FORMALISM FOR LEPTON HADRON SCATTERING. Apr 1970. 51p.	I.J. Muzinich, Ling-Lie Wang (Brookhaven); Jiunn-Ming Wang (SUNY, Stony Brook)
CALT-68-263	ET	DISTRIBUTIONS OF CHARGED PIONS. Jun 1970. 8p.	J. Horn, R. Silver (Cal Tech)
CERN-70-14	ETR	K-L-3 FORM FACTORS; REVIEW OF THE THEORY. REVIEW OF EXPERIMENTAL RESULTS. May 1970. 92p.	M.K. Gaillard (Orsay, LPTHE); L.M. Chounet (Orsay, LAL)
CERN-TH-1158	T	EXTRAPOLATION OF SCATTERING AMPLITUDES. 4. EXTRAPOLATION OFF A PART OF THE BOUNDARY TO THE REST OF THE BOUNDARY. May 1970. 36p.	P. Presnajder (Comenius U.); J. Pisut (CERN)
CERN-TH-1160	T	DUAL MULTIPARTICLE THEORY. May 1970. 112p.	V. Alesandrini, D. Amati, M. Le Bellac, D. Olive (CERN)
CERN-TH-1162	T	COULOMB CORRECTIONS TO ADLER'S PCAC TEST IN NUCLEI. May 1970. 24p.	O. Nachtmann (CERN)
CERN-TH-1164	T	BUILDING MESON BARYON SCATTERING AMPLITUDES FROM DUALITY. May 1970. 7p.	P.R. Auvil, F. Halzen, C. Michael (CERN)
CERN-TH-1165	ET	SIMPLE s HELICITY STRUCTURE AT THE NUCLEON-NUCLEON VERTEX OF THE POMERON AND MESON EXCHANGES: EXPERIMENTAL EVIDENCE AND CONNECTIONS WITH DUALITY. May 1970. 14p.	R. Odorico (CERN); R. Garcia (Centro Atomico Bariloche); C.A. Garcia-Canal (La Plata U.)
CERN-TH-1166	T	A PARTON VIEW ON DUAL AMPLITUDES. May 1970. 9p.	H.B. Nielsen (NORDITA); P. Olesen (CERN)
CERN-TH-1175	T	ON THE MASS DEPENDENCE OF MOMENTUM TRANSFER DISTRIBUTIONS IN DIFFRACTION DISSOCIATION. Jun 1970. 12p.	H. Satz (CERN)
Print-70-1387 (CLARKSON COLL.)	E	SCATTERING OF HIGH ENERGY PIONS ON ELECTRONS. n.d. 15p.	F.M. Waterman, P.J. McNulty (Clarkson Coll. Tech.)
COO-1545-76	T	UNITARIZED VENEZIANO MODEL WITH SATELLITES FOR pi pi SCATTERING. n.d. 32p.	David M. Scott, K. Tanaka, R. Torgerson (Ohio State U.)
CPT-57 (TEXAS)	T	NON-ASSOCIATIVITY OF THE OPERATORS IN THE CROSSING-SYMMETRIC BETHE-SALPETER EQUATIONS. Jun 1970. 30p.	Robert J. Yaes (Texas U.)
CTS-HE-70-3	T	SCALAR MESON DOMINANCE MODEL AND THE A1 epsilon pi SYSTEM. May 1970. 10p.	Mario Dal Cin (Miami U., Fla.)
CTS-LN-70-1	T	SPINORS IN HILBERT SPACE. May 1970. 100p.	P.A.M. Dirac (Miami U., Fla.)
DNPL/P-34	ET	REGGE CUT EFFECTS IN PION NUCLEON SCATTERING. Apr 1970. 23p.	B. Carreras, J.N.J. White (Daresbury)
Print-70-1388 (DREXEL)	T	GAUGE INVARIANCE AND VECTOR FIELDS. n.d. 9p.	I. Goldberg (Drexel U.)
Print-70-1391 (DURHAM)	T	DOUBLED TRAJECTORIES AND pi-minus p CHARGE EXCHANGE SCATTERING. Jun 1970. 7p.	Robert C. Johnson, Euan J. Squires (Durham U.)
Print-70-1385 (HEIDELBERG)	ET	DIFFRACTION DISSOCIATION AND THE 'A' REGION. n.d. 14p.	V.J. Stenger (Heidelberg U. and Hawaii U.)
IFT/70/7 (WARSAW)	T	MASS DEPENDENCE OF SLOPES IN DIFFERENTIAL CROSS SECTION AS A PROPERTY OF MODELS FOR PRODUCTION PROCESSES. Apr 1970. 14p.	S. Pokorski (Warsaw U.); M. Szeptycka (Warsaw, INR); L. Zieminski (Warsaw U.)
Print-70-1410 (INDIANA)	T	AMBIGUITIES IN THE MEANING OF DUALITY. n.d. 25p.	D.B. Lichtenberg, R.G. Newton, E. Predazzi (Indiana U.)
Print-70-1390 (IPN, MEXICO)	T	K2-zero ---> 2-pi + 2-gamma BY CURRENT ALGEBRA. n.d. 6p.	Lance Helko (IPN, Mexico); Lloyd C. Kannenberg (Lowell Tech. Inst. and Northeastern U.); Susan L. Kannenberg (Boston State Coll. and Northeastern U.)
Mat-2-1970	T	HIGH ENERGY PHOTOPRODUCTION OF PSEUDOSCALAR MESONS IN A QUARK MODEL. Jan 1970. 10p.	Ramesh Chand, A. Sundaram (Inst. Math. Sci., Madras)
MAT-3-1970	T	HIGH ENERGY PHOTOPRODUCTION OF VECTOR MESONS IN A QUARK MODEL. Feb 1970. 14p.	Ramesh Chand (Inst. Math. Sci., Madras)
MIT-CTP-131	T	EXACT CONSEQUENCES OF BROKEN O(4) SYMMETRY. 3. FACTORIZATION AND MASS DEPENDENCE. Apr 1970. 37p.	J.B. Bronzan, Uday P. Sukhatme (MIT, LNS)

*C = Computing E = Experimental I = Instrumentation R = Review T = Theoretical

Preprints in Particles and Fields

Report No.	*	Title	Authors
MIT-CTP-133	T	dpn AND tdn COUPLING CONSTANTS FROM FORWARD NEUTRON DEUTERON DISPERSION RELATIONS. May 1970. 25p.	M.P. Locher (MIT, LNS and SIN, Zurich)
MIT-CTP-134	T	DIRECT DETERMINATION OF THE METRIC FROM OBSERVED RED-SHIFTS AND DISTANCES. May 1970. 6p.	Steven Weinberg (MIT, LNS)
Print-70-1374 (NORDITA)	T	A DYNAMICAL MODEL FOR THE pi N RESONANCE S11(1525). Apr 1970. 44p.	Gosta Gustafson (NORDITA)
Print-70-1412 (PENN STATE)	T	AN INTERFERENCE MODEL SUGGESTED BY THE VENEZIANO AMPLITUDE. Feb 1970. 19p.	Emil Kazes, Kuo-hsiang Wang (Penn State U.)
PPAR-25	I	CALIBRATION OF A LARGE NEUTRON COUNTER USING THE PION NUCLEON CHARGE EXCHANGE REACTION. Apr 1970. 28p.	K.W. Chen, P.A. Crean, J.T. Dakin, M.G. Hauser (Princeton U.)
Print-70-1389 (PRINCETON)	T	BASIC OPERATORS OF THE DUAL-RESONANCE MODEL. May 1970. 42p.	David J. Gross, John H. Schwarz (Princeton U.)
PURC-2137-23	ET	CP VIOLATION AND REGENERATED K-zero-e-3 DECAY. Mar 1970. 107p. (Ph.D. Thesis)	Mark W. Strovink (Princeton U.)
Print-70-1400 (RUTGERS)	T	HARD-PION CURRENT ALGEBRA, UNITARIZATION, AND I=0 S-WAVE pi pi SCATTERING. Jun 1970. 11p.	Ronald Rockmore (Rutgers U.)
Print-70-1420 (RUTGERS)	ET	DEUTERON-DEUTERON SCATTERING AT 2.2- AND 7.9-GeV/c; THEORETICAL INTERPRETATION. n.d. 15p.	A.T. Goshaw, P.J. Oddone (Princeton U.); M.J. Bazin (Rutgers U.); C.R. Sun (SUNY Albany)
SLAC-PUB-766	E	DELTA PRODUCTION VIA gamma + p ---> DELTA + pi BY LINEARLY POLARIZED PHOTONS AT 2.8- AND 4.7-GeV. Jun 1970. 13p.	H.H. Bingham, W.B. Fretter, K.C. Moffeit, W.J. Podolsky, M.S. Rabin, A.H. Rosenfeld, R. Windmolders (UCRL, Berkeley); Joseph Ballam, G.B. Chadwick, R. Gearhart, Zaven G.T. Guiragossian, M. Menke, J.J. Murray, P. Seyboth, A. Shapira, Charles K. Sinclair, I.O. Skillicorn, Gunter E. Wolf (SLAC); R.H. Milburn (Tufts U.)
Print-70-1382 (TORONTO)	T	QUANTIZATION WITH THE ASYMPTOTIC CONDITION. n.d. 8p.	Robert E. Pugh (Toronto U.)
Print-70-1383 (TORONTO)	T	GENERATORS FOR INTERPOLATING FIELDS. n.d. 6p.	Robert E. Pugh (Toronto U.)
TPJU-8/70	T	EVALUATION OF THE ABSORPTION CORRECTIONS TO EXCHANGE MODELS. Apr 1970. 27p.	A. Bialas (Jagellonian U. and Cracow, INP); K. Zalewski (Cracow, INP)
TUL-30	T	ON QUANTUM MECHANICS AND STATISTICS OF IDENTICAL PARTICLES. Mar 1970. 30p. (In German)	Bodo Geyer (Karl-Marx U., Leipzig)
Print-70-1372 (UBC, VANCOUVER)	T	QUESTIONS ON pi PRODUCTION IN NUCLEI. n.d. 25p.	Douglas S. Beder (British Columbia U.)
UCRL-19709	ET	WEAK INTERACTIONS: ON THE EXISTENCE OF NEUTRAL LEPTONIC CURRENTS. Apr 1970. 69p. (Ph.D. Thesis)	Rolland P. Johnson (UCRL, Berkeley)
UCRL-19737	E	THE MESON SPECTRUM IN THE REACTION P + P ---> d + MM. Mar 1970. 183p. (Ph.D. Thesis)	Arthur Barry Wicklund (UCRL, Berkeley)
UCRL-19756	I	A MAGNETIC MONOPOLE DETECTOR UTILIZING SUPERCONDUCTING ELEMENTS. Apr 1970. 50p.	Luis W. Alvarez, Maurilio Antuna, Jr., Roscoe A. Byrns, Phillippe H. Eberhard, Robert E. Gilmer, Egon H. Hoyer, Ronald R. Ross, Hans H. Stellrecht, John D. Taylor (UCRL, Berkeley); Robert D. Watt (SLAC)
UCRL-19841	E	RADIATIVE PION-CAPTURE IN C-12. May 1970. 13p.	James A. Bistirlich, Kenneth M. Crowe, Anthony S.L. Parsons, P. Skarek, P. Truoel (UCRL, Berkeley)
UH-511-70-70	T	THEORETICAL ESTIMATE OF QUARK PRODUCTION CROSS SECTIONS AT ULTRA-HIGH ENERGIES. Jun 1970. 17p.	S.F. Tuan (Hawaii U.)
UT-48 (TOKYO)	T	THE THIRD ORDER CALCULATION IN A NON-POLYNOMIAL LAGRANGIAN THEORY. Jun 1970. 17p.	N. Sakai (Tokyo U.)
UT-49 (TOKYO)	T	A MODEL OF PARITY-CONSERVING NONLEPTONIC DECAYS OF HYPERONS. Jun 1970. 15p.	Tateaki Sasaki (Tokyo U.)
UT-70-4	T	THE REGGEIZATION OF THE BETHE-SALPETER SCATTERING AMPLITUDES. May 1970. 41p.	Hideki Matsumoto (Tokyo U.)
Print-70-1380 (WAYNE STATE)	T	UNIQUENESS OF THE RESPONSE TO THE VARIATION OF AN EXTERNAL GRAVITATIONAL FIELD. n.d. 26p.	J.C. Katzin, William B. Rolnick (Wayne State U.)
Print-70-1384 (YALE)	T	CROSSING SYMMETRY SUM RULES APPLIED TO BROWN AND GOBLE'S MODEL FOR pi pi SCATTERING. n.d. 23p.	Samuel Krinsky (Yale U.)

*C = Computing E = Experimental I = Instrumentation R = Review T = Theoretical

Anti - Preprints

First Author	PPF Nº	Partial Title, Date, Report Nº	Publication Info.
		"ANTI-PREPRINTS" is a periodic list of erstwhile preprints which have been published in current journals. The preprints may now be discarded, and reference made to the journal publication. The list is alphabetical by first author. Titles with no PPF number in column 2 are those which predate PPF "ANTI-PREPRINT CUMULATION NO. 3" has been mailed to those people who requested nos. 1 and 2. Any others who would like a copy may have one by sending a SELF ADDRESSED MAILING LABEL to: SLAC Library ATTN: ANTIPREPRINTS Box 4349 Stanford, Calif., U.S.A. 94305 REQUESTS NOT ACCOMPANIED BY A SELF-ADDRESSED MAILING LABEL WILL NOT BE FILLED.	
Abers, Ernest S. & Vigdor L.Teplitz	69-28	PI-rho SCATTERING IN THE VENEZIANO MODEL. Jun 1969. 48p. <MIT-CTP-92>	Phys.Rev.D1:624,1970
Amati,D., et al.	69-45	TWISTING INVARIANT FACTORIZATION OF MULTIPARTICLE DUAL AMPLITUDE. Nov 1969. 23p. <CERN-TH-1102>	Nuovo Cim.66A:315,1970
Amati,D., et al.	70-04	THE TWISTING OPERATOR IN MULTI-VENEZIANO THEORY. Dec 1969. 19p. <CERN-TH-1118>	Nuovo Cim.66A:831,1970
Anisovich,V.V. & O.A.Khrustalev	69-27	TWO-PARTICLE REACTIONS IN THE ORFAR REGION AT HIGH ENERGIES. Feb 1969. 14p. <IFVE-69-17> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:479,1970 (IN ENGLISH)
Antipov, Yu.M., et al.	69-11	SEARCH OF THE PARTICLES WITH A FRACTIONAL CHARGE (QUARKS) AT THE 70 GeV ACCELERATOR (IHEP). Dec 1968. 20p. <IFVE-68-72> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:199,1970 (also Phys.Lett.29B:245,1969) (IN ENGLISH)
Asbury,J.G., et al.	70-06	COSMIC RAY MUON SPECTRA NEAR 1000 GeV AT LARGE ZENITH ANGLES. Dec 1969. 26p. <ANL/HEP-6925>	Nuovo Cim.66B:169,1970
Baier,V.N. & V.S.Fadin	69-26	ON THE THEORY OF PION ELECTROMAGNETIC FORM FACTOR. n.d. 19p. <IYF-69-500> (IN RUSSIAN)	Nuovo Cim.66A:693,1970 (IN ENGLISH)
Balachandran,A.P. & Maurice L.Blackmon	70-22	INTEGRAL INEQUALITIES FOR THE s- AND p-WAVE pi pi PARTIAL-WAVE AMPLITUDES DUE TO CROSSING SYMMETRY AND POSITIVITY. Mar 1970. 7p. <NYO-3399-223>	Phys.Lett.31B:655,1970
Ball,J.S., et al.	70-12	PHENOMENOLOGICAL IMPLICATIONS OF COLLIDING REGGE POLES AND CUTS. n.d. 18p. <PRINT-70-558(Utah)>	Phys.Lett.31B:583,1970
Ballam,J., et al.	70-13	CONSERVATION OF S-CHANNEL HELICITY IN rho-zero PHOTO-PRODUCTION. Apr 1970. 21p. <SLAC-PUB-728>	Phys.Rev.Lett.24:960,1970
Barashenkov,V.S., et al.	69-13	INELASTIC INTERACTIONS OF HIGH-ENERGY DEUTERONS WITH ATOMIC NUCLEI. May 1968. 15p. <JINR-P2-4346> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:439,1970 (IN ENGLISH)
Barashenkov,V.S., et al.	69-07	MODIFICATION OF THE INELASTIC PARTICLE-NUCLEUS INTER-ACTION MECHANISM IN THE ENERGY REGION T-approximately-equal-to 5 GeV. Feb 1969. 10p. <JINR-P2-4302> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:436,1970 (IN ENGLISH)
Barbieri,R., et al.	70-17	ON THE FOURTH ORDER RADIATIVE CORRECTIONS TO THE ELECTRON-PHOTON VERTEX. Mar 1970. 5p. <PRINT-70-896(Pisa)>	Nuovo Cim.Lett.3:588,1970
Barger,V., et al.	69-09	BARYON EXCHANGES FROM u-CHANNEL FINITE ENERGY SUM RULES. Feb 1969. 26p. <COO-881-221>	Phys.Rev.185:1852,1969 ERRATUM-ibid.D1:374,1970
Barger,V. & R.J.N.Phillips	70-08	UNCONVENTIONAL ASYMPTOTICS. Jan 1970. 12p. <COO-881-262>	Phys.Lett.31B:643,1970
Barish,S.J., et al.	69-13	THE REACTION pi-minus + p --> pi-minus + pi-zero + p AT 8 GeV/c. n.d. 27p. <PRINT-69-902(Penn U.)>	Phys.Rev.184:1375,1969 ERRATUM-ibid.D1:375,1970
Barut,A.O., et al.		CROSSING SYMMETRY IN THE O(4,2) FORMULATION OF THE DIRAC THEORY. Oct 1968. 20p. <IC/68/96>	Phys.Rev.D1:536,1970
Barut,A.O. & G.J.Komen	69-06	ON THE REALIZATION OF CURRENT ALGEBRA COMMUTATION RE-LATIONS ON THE SPACE OF SOLUTIONS OF FINITE OR INFIN-ITE COMPONENT WAVE EQUATIONS. Jan 1969. 16p. <IC/69/8>	Phys.Rev.D1:418,1970
Barut,A.O. & J.W.Moffat	69-35	A SIMPLY FACTORIZABLE N-POINT AMPLITUDE. Sep 1969. 16p. <PRINT-69-2423(Colorado)>	Phys.Rev.D1:532,1970
Batusov,Yu.A., et al.	69-09	EXPERIMENTAL INVESTIGATION OF TWO-AND THREE-PARTICLE REACTIONS WITH LI-8 AND B-8 PRODUCTION IN SLOW pi-minus MESON CAPTURE BY C-12, N-14, O-16 NUCLEI. Feb 1969. 10p. <JINR-P1-4309> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:204,1970 (IN ENGLISH)

Anti - Preprints

First Author	PPF Nº	Partial Title, Date, Report Nº	Publication Info.
Batusov,Yu.A., et al.	69-31	PI-plus MESON PRODUCTION IN p p INTERACTION AT 6600 MeV. May 1969. 19p. <JINR-P1-4491> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:465,1970 (IN ENGLISH)
Bender,I., et al.	70-16	DUALITY AND FIXED POLES IN PION PHOTOPRODUCTION. Apr 1970. 12p. <PRINT-70-803(Heidelberg)>	Nuovo Cim.Lett.3:625,1970
Biggs,P.J., et al.	70-21	MEASUREMENT OF THE MAGNITUDE AND PHASE OF rho-omega INTERFERENCE IN pi-plus pi-minus PHOTOPRODUCTION FROM CARBON. Mar 1970. 13p. <DNPL/P-29>	Phys.Rev.Lett.24:1201,1970
Biggs,P.J., et al.	70-21	OBSERVATION OF rho-omega INTERFERENCE IN THE PHOTO-PRODUCTION OF ELECTRON POSITRON PAIRS FROM CARBON AND A MEASUREMENT OF THE rho-omega PHASE. Mar 1970. 13p. <DNPL/P-28>	Phys.Rev.Lett.24:1197,1970
Bilenkii,S.M., et al.	69-23	FORMATION AND DECAY OF (mu-plus, mu-minus) ATOMS. May 1969. 8p. <JINR-P2-4438> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:469,1970 (IN ENGLISH)
Bingham,H.H., et al.	70-13	THE REACTION gamma + p --> p + rho-zero WITH LINEARLY POLARIZED PHOTONS AT 2.8 AND 4.7 GeV: CROSS SECTIONS AND THE rho-zero MASS SHIFT. Apr 1970. 15p. <SLAC-PUB-727>	Phys.Rev.Lett.24:955,1970
Bogolubov,N.N.,Jr.	69-11	ABOUT MINIMAX PRINCIPLE FOR SOME MODEL PROBLEMS OF STATISTICAL PHYSICS. Nov 1968. 11p. <ITF-68-81>	Sov.J.Nucl.Phys.10:243,1970
Boright,J.P., et al.	70-06	THE REACTION pi-minus + p --> n + pi-zero IN THE BACK-WARD HEMISPHERE AT 6 GeV/c. Jan 1970. 11p. <CLNS-96>	Phys.Rev.Lett.24:964,1970
Buchanan,C.D.	69-44	THE CASE FOR Im(DELTA-S = DELTA-Q (A-A) AS THE ONLY SOURCE OF CP-VIOLATION. n.d. 11p. <UCLA-1045>	Nuovo Cim.Lett.3:529,1970
Bugg,W.M., et al.	70-18	HYPERFRAGMENT PRODUCTION FOLLOWING SLOW SIGMA-minus CAPTURE IN LIQUID NEON. n.d. 11p. <PRINT-70-918(Tennessee)>	Phys.Lett.31B:595,1970
Burnett,T.H. & John H.Schwarz	69-39	INFINITE PRODUCTS AND CONVERGENCE PROPERTIES OF CER-TAIN REGGEIZED FEYNMAN DIAGRAMS. Aug 1969. 32p. <PRINT-69-2615(Princeton)>	Phys.Rev.D1:423,1970
Cabibbo,N. & L.Maiani	69-20	ON THE ORIGIN OF THE WEAK INTERACTION ANGLE, II. May 1969. 47p. <ISS 69/18>	Phys.Rev.D1:707,1970
Caiucci,G. & R.Jengo	70-11	A MODEL FOR VERY INELASTIC ELECTRON PROTON SCATTER-ING. n.d. 11p. <PRINT-70-514(Trieste)>	Nuovo Cim.Lett.3:455,1970
Campbell,J.A. & Anthony C.Hearn	69-41	SYMBOLIC ANALYSIS OF FEYNMAN DIAGRAMS BY COMPUTER. Aug 1969. 75p. <ITP-333(Stanford)> <CPT-4(Texas)>	J.Comp.Phys.5:280,1970
Carmeli, Moshe	69-24	CONSTRAINTS IN ELECTRODYNAMICS AND QUANTIZATION OF THE ELECTROMAGNETIC FIELD WITHOUT CHARGES. n.d. 17p. <PRINT-69-1684(ARL,WPAFB,ohio)>	Nuovo Cim.67B:103,1970
Chavda,L.K.	70-15	FINITE TRANSFORM SUM RULES. n.d. 7p. <PRINT-70-698(Purdue)>	Nuovo Cim.Lett.3:622,1970
Cheng, Hung & Tai Tsun Wu	69-31 69-30	DIFFRACTION SCATTERING FOR INELASTIC PROCESSES, I & II n.d. 26p. <PRINT-69-2073 & PRINT-69-2045(MIT)> *Two preprints seem to be combined in one Jnl. paper*	Phys.Rev.D1:459,1970
Cheng, Hung & Tai Tsun Wu	69-37	ELASTIC SCATTERING PROCESSES IN SCALAR ELECTRODYNAMICS AT HIGH ENERGIES I. n.d. 24p. <PRINT-69-2512(MIT)>	Phys.Rev.D1:467,1970
Cheng, Hung & Tai Tsun Wu	69-44	HIGH ENERGY PRODUCTION PROCESSES IN QUANTUM ELECTRO-DYNAMICS. n.d. 19p. <PRINT-69-2928(MIT)>	Phys.Rev.D1:456,1970
Cornwall, John M., et al.	70-06	TWO ASYMPTOTIC SUM RULES FOR ELECTROPRODUCTION AND PHOTOPRODUCTION. n.d. 8p. <PRINT-70-232(UCLA)>	Phys.Rev.Lett.24:1141,1970
Dash, Jan W., et al.	69-48	A RESONANCE CROSSING-SYMMETRIC CONFLUENT-HYPERGEOMET-RIC FUNCTION MODEL WITH COMPLEX TRAJECTORIES. Dec 1969. 30p. <RLO-1388-556rev.>	Nuovo Cim.67A:54,1970
de Mottoni,P. & H.Genz	70-11	FIELD PRODUCTS AT SHORT DISTANCES. n.d. 18p. <PRINT-70-492(Hamburg)>	Nuovo Cim.67B:1,1970
Drago,F., et al.	70-17	LACK OF DIP SYSTEMATICS IN STRONG CUT MODELS. n.d. 9p. <PRINT-70-903(Rutherford)>	Phys.Lett.31B:647,1970
Drummond,I.T.	70-01	THE SYMMETRICAL VERTEX AND THE TREE STRUCTURE OF DUAL AMPLITUDES. Oct 1969. 32p. <DAMTP-69/36>	Nuovo Cim.67A:71,1970
Dubin, Daniel A.		THE GROUP THEORETICAL STRUCTURE OF FREE QUANTUM FIELDS IN TWO DIMENSIONS. n.d. 25p. <ICTP/67/37>	Nuovo Cim.67B:39,1970
Durr,H.P. & E.Seiler	70-02	ANALYTICITY AND DIPOLE REGULARIZATION. Nov 1969. 17p. <PRINT-70-49(MPI,Munich)>	Nuovo Cim.66A:734,1970
Englert,F., et al.	69-47	MASS QUANTIZATION CONDITIONS FROM INFINITE SUPERCON-VERGENCE AND CHIRAL SYMMETRY. n.d. 20p. <PRINT-69-3081(Brussels)>	Nuovo Cim.66A:845,1970

Anti - Preprints

First Author	PPF Nº	Partial Title, Date, Report Nº	Publication Info.
Feldman,G. & P.T.Matthews	69-30	THE HIGH ENERGY LIMIT OF PRODUCTION AMPLITUDES. Jun 1969. 40p. <ICTP/68/40>	Phys.Rev.D1:523,1970
Ferrara,S. & G.Mattioli	70-19	ON THE REGGEIZED MULTIPERIPHERAL MODEL. Dec 1969. 19p. <INFN-ROME-245>	Nuovo Cim.67A:1,1970
Flamm,D. & W.Majerotto	69-39	REFINED QUARK MODEL OF WEAK HADRON DECAYS. Aug 1969. 15p. <HEP-12(Vienna)>	Nuovo Cim.66A:797,1970
Fleming, Gordon N.	70-02	THE DESCRIPTION OF PARTICLES IN SCALAR HYPERPLANE DEPENDENT FIELD THEORY I. THE GENERAL FORMALISM. n.d. 26p. <PRINT-70-21(Penn State)>	Phys.Rev.D1:542,1970
Frazer,W.R. & C.H.Mehta	69-32	A SIMPLIFIED MULTI-REGGE MODEL AT t not-equal-to 0. n.d. 19p. <UCSD-10P10-65>	Phys.Rev.D1:696,1970
Fried,H.M.	69-30	FIELD THEORETIC MODEL FOR HIGH ENERGY SCATTERING II. EFFECTIVE REGGE EXCHANGE IN pi N SCATTERING. n.d. 34p. <NYO-2262TA-207> *Title changed in Journal*	Phys.Rev.D1:596,1970
Fubini,S. & G.Veneziano	70-11	DUALITY IN OPERATOR FORMALISM. Dec 1969. 33p. <MIT-CTP-114>	Nuovo Cim.67A:29,1970
Genz,H.	69-31	ON BARYON SPECTRAL FUNCTION SUM RULES. Jul 1969. 15p. <UCRL-19243>	Phys.Rev.D1:659,1970
Gleeson,A.M. & E.C.G.Sudarshan	69-31	PHYSICAL INTERPRETATION OF COMPLEX ENERGY NEGATIVE METRIC THEORIES. Jul 1969. 44p. <NYO-3399-200>	Phys.Rev.D1:474,1970
Gottlieb,H.P.W.	70-18	SYMMETRY BREAKING IN LAGRANGIAN THEORIES OF CURRENT ALGEBRA. Mar 1970. 6p. <DAMTP-70/10>	Nuovo Cim.Lett.3:693,1970
Green,M.B.	69-33	TWO BODY UNITARITY AND THE ASYMPTOTIC DUALITY SERIES. Jul 1969. 20p. <PRINT-69-2278(Cambridge)>	Phys.Rev.D1:450,1970
Gross, David J., et al.	70-14	THE PRIMITIVE GRAPHS OF DUAL-RESONANCE MODELS. Mar 1970. 13p. <PRINT-70-635(Princeton)>	Phys.Lett.31B:592,1970
Hartle, James B.	69-38	LONG RANGE WEAK FORCES AND COSMOLOGY. n.d. 15p. <PRINT-69-2571(UC,Santa Barbara)>	Phys.Rev.D1:394,1970
Holder,M.		DEPENDENCE OF NEUTRINO CROSS SECTIONS ON THE ATOMIC WEIGHT OF THE TARGET. n.d. 8p. <PITHA-29>	Nuovo Cim.Lett.3:445,1970
Honerkamp,J.	69-38	SPONTANEOUS BREAKDOWN OF SU(3) x SU(3) TO U(2). Aug 1969. 31p. <BONN-2-66>	Nuovo Cim.66A:767,1970
Kantor,P.B.		A POSSIBLE DIFFICULTY WITH THE "EIKONAL PICTURE" OF HIGH ENERGY INTERACTIONS. n.d. 8p. <COO-1573-66>	Phys.Rev.Lett.24:1081,1970
Kemp,M.A.R. & D.L.Ward	70-17	Pi-minus p ELASTIC SCATTERING FROM 1225 TO 2070 MeV/c COMPARED WITH PHASE SHIFT ANALYSIS. Feb 1970. 15p. <DNPL/P-25>	Phys.Lett.31B:613,1970
Khrustalev,O.A., et al. *V.I.Savrin 1st auth.in Jnl.*	69-21	SEMIPHENOMENOLOGICAL ANALYSIS OF THE SCATTERING AMPLITUDE AT HIGH ENERGIES II. Mar 1969. 16p. <IFVE-69-23> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:495,1970 (IN ENGLISH)
Kim,C.W., et al.	69-23	RENORMALIZATION OF THE AXIAL-VECTOR VERTEX IN SPINOR ELECTRODYNAMICS. n.d. 38p. <PRINT-69-1598(Johns Hopkins U.)>	Phys.Rev.D1:434,1970
King, Marcia J. & P.K.Kuo	69-09	KINEMATIC CONSTRAINT RELATIONS FOR HELICITY AMPLITUDES. n.d. 37p. <PRINT-69-594(Johns Hopkins)> *Title changed in Journal*	Phys.Rev.D1:442,1970
Klein, Stanley	69-18	A GENERAL SOLUTION FOR REGGE RESIDUES AND TRAJECTORIES. n.d. 49p. <PRINT-69-1262(Claremont)>	Phys.Rev.D1:609,1970
Klems,J.H., et al.	70-22	A LIMIT ON THE K-plus --> pi-plus + neutrino + antineutrino DECAY RATE. Mar 1970. 12p. <UCRL-19469>	Phys.Rev.Lett.24:1086,1970
Kosterlitz,J.M. & D.A.Wray	70-19	THE GENERAL N POINT VERTEX IN A DUAL MODEL. Feb 1970. 16p. <PRINT-70-974(Turin)>	Nuovo Cim.Lett.3:491,1970
Kottler,H.	69-48	THE POMERANCHUK THEOREM AND THE POMERANCHUK-OKUN RULE. Sep 1969. 12p. <NYO-3829-39>	Nuovo Cim.67A:88,1970
Kudryavtsev,A.E.	69-20	ON THEORY OF RESONANCE REACTIONS IN THE REGION OF OVERLAPPING LEVELS. n.d. 12p. <ITEP-659> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:179,1970 (IN ENGLISH)
Leutrup,B.E., et al.	70-14	CONFIRMATION OF A NEW THEORETICAL VALUE FOR THE LAMB SHIFT. Mar 1970. 7p. <CERN-TH-1140>	Phys.Lett.31B:577,1970
Logan,R.K. & D.P.Roy	69-48	DUALITY AND NONET DEGENERACY. n.d. 10p. <PRINT-69-3154(Toronto)>	Nuovo Cim.Lett.3:517,1970
Lomon, Earle L.	69-41	COUPLED CHANNEL AMPLITUDES: PHYSICAL EFFECTS AND ANALYTIC STRUCTURE. Aug 1969. 65p. <MIT-CTP-102>	Phys.Rev.D1:549,1970
Lukierski,J.	70-01	COMPLEX LEE-WICK POLES AS CDD GHOSTS. Dec 1969. 12p. <CERN-TH-1114>	Nuovo Cim.66A:807,1970

Anti - Preprints

First Author	PPF Nº	Partial Title, Date, Report Nº	Publication Info.
Maltsev,V.M.		SUM RULES FOR THE PROCESSES OF MULTIPLE PION PRODUCTION BY HADRONS. Nov 1968. 10p. <JINR-P2-4169> (IN RUSSIAN)	Sov.J.Nucl.Phys.10:215,1970 (IN ENGLISH)
Moen,I.O. & J.W.Moffat	69-40	Anti-p + n --> 3 pi, K --> 3 pi AND eta --> 3 pi DECAYS IN A CROSSING SYMMETRIC REGGE POLE MODEL. Oct 1969. 18p. <PRINT-69-2711(Toronto)>	Nuovo Cim.Lett.3:473,1970
Nagylaki, Thomas	69-32	ELECTROMAGNETIC MASS DIFFERENCES OF THE PSEUDOSCALAR MESON AND BARYON OCTETS. Aug 1969. 46p. <CALT-68-212>	Phys.Rev.D1:637,1970
Narodetskii,I.M.		HILBERT-SCHMIDT EXPANSION FOR TWO-PARTICLE AMPLITUDE POTENTIAL SCATTERING. 1968. 35p. <ITEP-621> (IN RUSSIAN)	Sov.J.Nucl.Phys.9:636,1969 (IN ENGLISH)
Nussinov,S. & J.Rosner	69-31	UNIVERSAL ISOVECTOR CURRENT WITH MANY 1-minus POLES. n.d. 10p. <BNL-13902>	Phys.Rev.D1:656,1970
Oehme, Reinhard	70-12	COMPLEX POMERANCHUK TRAJECTORIES. Mar 1970. 12p. <EFI-70-16>	Phys.Lett.31B:573,1970
Omnes,R.	69-32	THE EFFECT OF ANNIHILATION ON MATTER-ANTIMATTER SEPARATION. Jul 1969. 9p. <LPTHE-TH-69/48>	Phys.Rev.D1:723,1970
Parisi,G. & M.Testa	70-19	GENERATING FUNCTIONALS, WARD IDENTITIES AND SCALAR MESONS. Sep 1969. 14p. <INFN-ROME-242>	Nuovo Cim.67A:13,1970
Pestieau, Jean & Hidezumi Terazawa	70-10	NEW CONSTRAINTS IN HIGH ENERGY ELECTRON-POSITRON ANNIHILATION INTO HADRONS. Feb 1970. 10p. <CLNS-97>	Phys.Rev.Lett.24:1149,1970
Plahte,E.	69-46	SYMMETRY PROPERTIES OF DUAL TREE GRAPH N POINT AMPLITUDES. Nov 1969. 33p. <CERN-TH-1096>	Nuovo Cim.66A:713,1970
Rabin, Monroe S., et al.	69-47	EVIDENCE AGAINST A-2 PRODUCTION IN HIGH ENERGY K-plus p INTERACTIONS. Oct 1969. 16p. <UCRL-19362>	Phys.Rev.Lett.24:925,1970
Rafanelli, Kenneth	69-36	OBSERVABLE PARTICLE MOTIONS. n.d. 10p. <PRINT-69-2675(Queens Coll)>	Nuovo Cim.67A:48,1970
Ranft,J.	70-24	SECONDARY PARTICLE PRODUCTION ACCORDING TO THE THERMODYNAMICAL MODEL AND NEW EXPERIMENTAL DATA. Mar 1970. 13p. <TUL-28>	Phys.Lett.31B:529,1970
Restignoli,M. & G.Violini	70-09	DO K MESONS VIOLATE THE POMERANCHUK THEOREM? Feb 1970. 10p. <INFN-ROME-257>	Phys.Lett.31B:533,1970
Richards,W.Bruce, et al.	69-21	PRODUCTION AND NEUTRAL DECAY OF THE ETA MESON IN pi-minus p COLLISIONS. May 1969. 43p. <UCRL-18879>	Phys.Rev.D1:10,1970
Rockmore, Ronald	69-35	PION POLE MECHANISM AND THE APPLICATION OF THE VENEZIANO MODEL TO K-2-zero --> pi + pi - gamma DECAY. Aug 1969. 15p. <PRINT-69-2433(Rutgers)>	Phys.Rev.D1:226,1970
Rockmore, Ronald	70-01	SIMPLE APPROACH TO UNITARIZATION IN HARD-MESON CALCULATIONS. n.d. 8p. <PRINT-69-3189(Rutgers)>	Phys.Rev.Lett.24:541,1970 ERRATUM-ibid.:970,1970
Ryan,C.	69-33	ELECTROMAGNETIC CONTRIBUTIONS TO THE CHARGE ASYMMETRY IN THE SEMILEPTONIC DECAYS OF NEUTRAL KAONS. Aug 1969. 22p. <IC/69/92>	Phys.Rev.D1:299,1970
Shabalin,E.P.		ON SOME MODEL OF RENORMALIZABLE WEAK INTERACTION. Aug 1968. 22p. <ITEP-635> (IN RUSSIAN)	Sov.J.Nucl.Phys.9:615,1969 (IN ENGLISH)
Singer, Paul	69-42	DERIVATION OF KSFR-TYPE RELATIONS. n.d. 18p. <PRINT-69-2843(Northwestern)>	Phys.Rev.D1:321,1970
Singer, Paul	69-40	RADIATIVE X-zero DECAYS IN BROKEN SU(3). Sep 1969. 8p. <DESY 69/35>	Phys.Rev.D1:86,1970
Spector, Richard M.	69-36	A REGGE CLASSIFICATION OF THE CASCADE RESONANCES. Sep 1969. 12p. <PRINT-69-2458(Wayne State)>	Phys.Rev.D1:365,1970
Suen,K.F., et al.	69-22	SIX-PRONGED pi-minus p INTERACTIONS AT 6 GeV/c. n.d. 48p. <PRINT-69-1498(Indiana)>	Phys.Rev.D1:54,1970
Thompson, Richard H.	69-15	A THREE-DIMENSIONAL BETHE-SALPETER EQUATION APPLIED TO THE NUCLEON-NUCLEON INTERACTION. n.d. 20p. <PRINT-69-1100(Florida)>	Phys.Rev.D1:110,1970
Vasavada, Kashyap V.	69-34	VENEZIANO MODEL WITH SECONDARY TERMS FOR PION-PION SCATTERING. Jun 1969. 25p. <PRINT-69-2323(Connecticut)>	Phys.Rev.D1:88,1970
Vasilevskii,I.M., et al.		PATHS AND IONIZATION LOSSES OF THE PROTON ENERGY IN DIFFERENT MATTERS. Sep 1968. 27p. <JINR-P1-4081> (IN RUSSIAN)	Sov.J.Nucl.Phys.9:583,1969 (IN ENGLISH)
von Hippel, Frank & Jae Kwan Kim	69-42	THE NATURE OF SU(3) x SU(3) SYMMETRY BREAKING - RESULTS FROM A SYSTEMATIC TEST OF THE SOFT MESON THEOREMS. Sep 1969. 43p. <ITP-342(Stanford)>	Phys.Rev.D1:151,1970
Walsh, Thomas F. & Earl A. Peterson	69-46	AXIAL-VECTOR SUM RULES FOR He-3. Jul 1969. 36p. <COO-1764-38>	Phys.Rev.D1:274,1970

APPENDIX D: SLAC Publications List (Sample Pages)

SLAC PUBLICATIONS

NUMERICAL LIST

Reports
Preprints and Reprints
Translations

SUBJECT LIST

Reports, Preprints, Reprints

AUTHOR LIST

Reports, Preprints, Reprints

February 1, 1969

Technical Information Department
Stanford Linear Accelerator Center
Stanford University
Stanford, California

NUMERICAL LIST

February 1, 1969

Stanford Linear Accelerator Center
Stanford University
Stanford, California

SLAC REPORTS

SLAC-1
TWO-MILE ACCELERATOR PROJECT; QUARTERLY STATUS REPORT,
1 APR TO 30 JUN 1962.
STANFORD LINEAR ACCELERATOR CENTER, CALIF . Jul 1962. 74p.

SLAC-2
DISCUSSION OF FOCUSING REQUIREMENTS FOR THE STANFORD TWO-MILE ACCELERATOR.
Richard H. Helm. Aug 1962. 39p.

SLAC-3
SHOWER DEVELOPMENT AND HEATING IN THE WAVE-GUIDE STRUCTURE WITH AN 800 MeV
ELECTRON BEAM.
Joseph K. Cobb, J.J. Muray. Jul 1962. 33p.

SLAC-4
ADIABATIC APPROXIMATION FOR DYNAMICS OF A PARTICLE IN THE FIELD OF A
TAPERED SOLENOID.
Richard H. Helm. Aug 1962. 17p.

SLAC-5
SOME ASPECTS OF THE PROSPECTIVE EXPERIMENTAL USE OF THE STANFORD TWO-MILE
ACCELERATOR.
William Chinowsky, John W. DeWire, D.B. Lichtenberg, G. Masek,
J.J. Murray, Martin L. Perl, Melvin Schwartz, J. Tinlot, G. Trilling.
Summer Study Group, SLAC, 1962.

SLAC-5-A
PHOTON BEAM FROM PROJECT M ACCELERATOR.
John W. DeWire. Aug 1962. 19p.
pt.A of SLAC-5, p.1-19.

SLAC-5-B
CONJECTURES ON THE EFFECTS OF REGGE POLES ON DRELL PROCESSES.
D.B. Lichtenberg. Aug 1962. 9p.
Pt.B of SLAC-5, p. 20-28.

SLAC-5-C
A PROPOSED METHOD TO SEARCH FOR INTERMEDIATE BOSONS AND HEAVY LEPTONS.
Melvin Schwartz. Aug 1962. 3p.
Pt.C of SLAC-5, p. 29-31.

SLAC-5-D
KINEMATIC CALCULATIONS TO DETERMINE YIELDS OF PARTICLES ARISING FROM THE
DECAYS OF SHORT-LIVED INTERMEDIATE STATES.
G. Trilling. Aug 1962. 8p.
pt. D of SLAC-5, p. 32-9.

SLAC-5-E
THE USE OF HYDROGEN BUBBLE CHAMBERS AT SLAC.
G. Trilling. Aug 1962. 39p.
Pt. E of SLAC-5, p. 40-78.

SLAC REPORTS (Cont.)

SLAC-5-F
SOME CONSIDERATIONS ON BUBBLE CHAMBER EXPERIMENTS WITH M.
William Chinowsky. Aug 1962. 11p.
pt. F of SLAC-5, p. 79-89.

SLAC-5-G
STRONG INTERACTION PHYSICS WITH SPARK CHAMBERS.
Martin L. Perl. Aug 1962. 43p.
pt.G of SLAC-5, p. 90-132.

SLAC-5-H
SPARK CHAMBER DETECTION SYSTEM FOR 3-BeV STORAGE RING.
Martin L. Perl. Aug 1962. 21p.
pt. H of SLAC-5, p. 133-64.

SLAC-5-I
A STORAGE RING FOR 10-BeV MU MESONS.
J. Tinlot. Aug 1962. 28p.
pt. I of SLAC-5, p. 165-92.

SLAC-5-J
mu-BEAMS WITH M AND THEIR APPLICATION TO mu-p ELASTIC SCATTERING EXPERIMENTS.
G. Masek. Aug 1962. 29p.
pt. J of SLAC-5, p. 193-221.

SLAC-5-K
MASS ANALYSIS AT HIGH ENERGY.
J.J. Murray. Aug 1962. 15p.
pt. K of SLAC-5, p. 222-36.

SLAC-6
THE PRINCIPLE OF DESIGN OF MAGNETIC MOMENTUM SLITS.
Joseph Ballam. Oct 1962. 37p.

SLAC-7
DESIGN AND FABRICATION OF THE ACCELERATING STRUCTURE FOR THE STANFORD TWO-MILE ACCELERATOR.
Arnold L. Eldredge, G.A. Loew, Richard B. Neal. Nov 1962. 82p.

SLAC-8
TWO-MILE ACCELERATOR PROJECT; QUARTERLY STATUS REPORT,
1 JUL TO 30 SEP 1962.
STANFORD LINEAR ACCELERATOR CENTER, CALIF . Nov 1962. 86p.

SLAC-9
TRANSVERSE RADIATION SHIELDING FOR THE STANFORD TWO-MILE ACCELERATOR.
H.C. DeStaebler, Jr.. Nov 1962. 111p.

SLAC-10
TWO-MILE ACCELERATOR PROJECT; QUARTERLY STATUS REPORT,
1 OCT TO 30 DEC 1962.
STANFORD LINEAR ACCELERATOR CENTER, CALIF . Mar 1963. 80p.

SLAC REPORTS (Cont.)

SLAC-11
MISALIGNMENT AND QUADRUPOLE ERROR EFFECTS IN A FOCUSING SYSTEM FOR THE TWO-MILE ACCELERATOR.
Richard H. Helm. Jan 1963. 54p.

SLAC-12
SPACE-CHARGE LIMIT AND CONFINEMENT IN PARTICLE ACCELERATION WITH HIGH-INTENSITY LIGHT WAVES.
J.J. Muray. Jan 1963. 19p.

SLAC-13
HEAVY MESONS AND EXCITED BARYONS.
D.B. Lichtenberg. Mar 1963. 93p.

SLAC-14
OPTICAL PROPERTIES OF QUADRUPOLE MULTIPLETS FOR SECTOR FOCUSING IN THE TWO-MILE ACCELERATOR.
Richard H. Helm. Feb 1963. 31p.

SLAC-15
MISALIGNMENT AND QUADRUPOLE ERROR PROBLEMS AFFECTING THE CHOICE OF MULTIPLET TYPE FOR SECTOR FOCUSING OF THE TWO-MILE ACCELERATOR.
Richard H. Helm. Mar 1963. 51p.

SLAC-16
TWO-MILE ACCELERATOR PROJECT; QUARTERLY STATUS REPORT, 1 JAN TO 31 MAR 1963.
STANFORD LINEAR ACCELERATOR CENTER, CALIF . May 1963. 94p.

SLAC-17
INVESTIGATION OF TRAVELING-WAVE SEPARATORS FOR THE STANFORD TWO-MILE LINEAR ACCELERATOR.
Otto H. Altenmueller, Rudolf R. Larsen. Aug 1963. 45p.

SLAC-18
TWO-MILE ACCELERATOR PROJECT; QUARTERLY STATUS REPORT, 1 APR TO 30 JUN 1963.
STANFORD LINEAR ACCELERATOR CENTER, CALIF . Aug 1963. 111p.

SLAC-19
A BENDING MAGNET WITH NONSATURATING SHIMMING.
B. Hedin. Sep 1963. 31p.

SLAC-20
EFFECTS OF STRAY MAGNETIC FIELDS AND RF COUPLER ASYMMETRY IN THE TWO- MILE ACCELERATOR WITH SECTOR FOCUSING.
Richard H. Helm. Oct 1963. 23p.

SLAC-21
PION PHOTOPRODUCTION IN HYDROGEN AND BERYLLIUM.
Martial L. Thiebaux, Jr.. Nov 1963. 44p.

SUBJECT LIST

of

Reports, Preprints, and Reprints

Page

February 1, 1969

Stanford Linear Accelerator Center
Stanford University
Stanford, California

THEORETICAL PHYSICS

SLAC-5-B
CONJECTURES ON THE EFFECTS OF REGGE POLES ON DRELL PROCESSES.
D.B. Lichtenberg. Aug 1962. 9p.
Pt. B of SLAC-5, p. 20-28.

SLAC-13
HEAVY MESONS AND EXCITED BARYONS.
D.B. Lichtenberg. Mar 1963. 93p.

SLAC-21
PION PHOTOPRODUCTION IN HYDROGEN AND BERYLLIUM.
Martial L. Thiebaux, Jr.. Nov 1963. 44p.

SLAC-25, pt. 1-A
COMPARISON OF ESTIMATED PION FLUXES AT THE AGS WITH PION FLUXES EXPECTED AT SLAC.
D.B. Lichtenberg. Jan 1964. 19p.
pt. A of SLAC-25.

SLAC-25, pt. 1-B
A CHARGE AND ISOTOPIC SPIN ANALYSIS ON THE WEAK INTERACTIONS.
G. Feldman. Jan 1964. 11p.
pt. B of SLAC-25.

SLAC-25, pt. 1-M
PAIR PRODUCTION OF VECTOR BOSONS IN THE COULOMB FIELD OF A NUCLEUS.
J. Mathews. Jan 1964. 5p.
pt. M of SLAC-25.

SLAC-43
SPIN AND PARITY ANALYSIS IN TWO-STEP DECAY PROCESSES.
S. M. Berman, Maurice Jacob. May 1965. 34p.

SLAC-58
INFINITE LADDERS AND THE ALGEBRA OF INTEGRATED CURRENT COMPONENTS.
Haim Harari. Jan 1966. 13p.

SLAC-67
TABLES OF gamma SPECTRA FROM e-plus HYDROGEN ATOM COLLISIONS.
A.J. Dufner, S. Swanson, Yung-Su Tsai. Aug 1966. 132p.

SLAC-79
FADDEEV'S EQUATIONS FOR LOCAL POTENTIALS.
Thomas A. Osborn. Dec 1967. 125p.

SLAC-PUB-1
CLASSIFICATION OF MESONS IN A COMPOUND MODEL.
D. B. Lichtenberg.
Published in Nuovo Cim. 27, 860-7 (1963).

SLAC-PUB-5
NEUTRON-PROTON SCATTERING BELOW 20 MeV.
H. Pierre Noyes.
Published in Phys. Rev. 130, 2025-33 (1963).

THEORETICAL PHYSICS (Cont.)

SLAC-PUB-8
THE VECTOR THEORY OF STRONG INTERACTIONS AND ANTINUCLEON-NUCLEON ANNIHILATION.
S. M. Berman, Robert J. Oakes.
Published in Nuovo Cim. 29, 1329-37 (1963).

SLAC-PUB-9
A POSSIBLE METHOD TO OBTAIN EVIDENCE ON THE SPINS AND PARITIES OF EXCITED BARYONS.
D. B. Lichtenberg.
Published in Phys. Lett. 4, 73-4 (1963).

SLAC-PUB-10
COHERENT PRODUCTION AS A MEANS OF DETERMINING THE SPIN AND PARITY OF BOSONS.
S. M. Berman, S. D. Drell.
Published in Phys. Rev. Lett. 11, 220-4 (1963).

SLAC-PUB-11
PION PRODUCTION IN PION-PION COLLISIONS.
Martial L. Thiebaux, Jr..
Published in Phys. Rev. 131, 854-9 (1963).

SLAC-PUB-14
METHODS OF DETERMINING THE QUANTUM NUMBERS OF EXCITED MESON AND BARYON STATES.
D. B. Lichtenberg.
Published in Proc. of Athens Topical Conference on Recently Discovered Resonant Particles, Athens, Ohio, 1963. Athens, Ohio, Ohio U., 1963. p. 152-61.

SLAC-PUB-16
SPECULATIONS ON THE PRODUCTION OF VECTOR MESONS.
S. M. Berman, S. D. Drell.
Published in Phys. Rev. 133, B791-801 (1964).

SLAC-PUB-18
INTERMEDIATE BOSON PAIR PRODUCTION AS A MEANS FOR DETERMINING ITS MAGNETIC MOMENT.
S. M. Berman, Yung-Su Tsai.
Published in Phys. Rev. Lett. 11, 483-7 (1963).

SLAC-PUB-21
ELECTRODYNAMIC PROCESSES WITH NUCLEAR TARGETS.
S. D. Drell, J. D. Walecka.
Published in Ann. Phys. (N. Y.) 28, 18-33 (1964).

SLAC-PUB-24
LOW-ENERGY NEUTRON-NEUTRON SCATTERING PARAMETERS.
Myron Bander.
Published in Phys. Rev. 134, B1052-7 (1964).

THEORETICAL PHYSICS (Cont.)

SLAC-PUB-25
DETERMINATION OF THE PROTON-PROTON s-wave singlet SHAPE PARAMETER.
H. Pierre Noyes.
Published in Phys. Rev. Lett. 12, 171-6 (1964).

SLAC-PUB-26
ANGULAR CORRELATIONS IN PRODUCTION PROCESSES.
S. M. Berman, Robert J. Oakes.
Published in Phys. Rev. 135, B1034-40 (1964).

SLAC-PUB-27
UNIFICATION OF PHOTOPRODUCTION AND ELECTROPRODUCTION.
S.M. Berman.
Published in Phys. Rev. 135, B1249-54 (1964).

SLAC-PUB-29
THE INTERPRETATION OF LOW ENERGY NUCLEON-NUCLEON SCATTERING.
H. Pierre Noyes.
Published in Proceedings of Congress International de Physique Nucleaire, Paris, Jul 1964. Paris, Editions du CNRS, 1964.

SLAC-PUB-30
ESTIMATIONS OF HIGH-ENERGY PHOTO-PION PRODUCTION AT 0-degrees.
Uri Maor.
Published in Phys. Rev. 135, B1205-11 (1964).

SLAC-PUB-31
AN ITERATIVE SOLUTION OF THE N/D EQUATION.
Myron Bander.
Published in J. Math. Phys. 5, 1427-30 (1964).

SLAC-PUB-32
NUCLEAR OPTICAL MODEL FOR VIRTUAL PIONS.
John S. Bell.
Published in Phys. Rev. Lett. 13, 57-9 (1964).

SLAC-PUB-33
MESON EXCHANGE EFFECTS IN ELASTIC e-D SCATTERING.
Ronald J. Adler, S. D. Drell.
Published in Phys. Rev. Lett. 13, 349-52 (1964).

SLAC-PUB-34
OFF-SHELL CORRECTION IN PION PHOTOPRODUCTION.
Martial L. Thiebaux, Jr..
Published in Phys. Rev. Lett. 13, 29-32 (1964).

SLAC-PUB-35
ASYMMETRIC mu-PAIR PHOTOPRODUCTION AT SMALL ANGLES.
S. D. Drell.
Published in Phys. Rev. Lett. 13, 257-60 (1964).

THEORETICAL PHYSICS (Cont.)

SLAC-PUB-38
EFFECTS OF THE rho-NUCLEON TOTAL CROSS SECTION ON PHOTOPION PRODUCTION IN THE FORWARD DIRECTION.
S. M. Berman, Uri Maor.
Published in Nuovo Cim. 36, 483-7 (1965).

SLAC-PUB-39
BOUNDS ON PROPAGATORS, COUPLING CONSTANTS AND VERTEX FUNCTIONS.
S. D. Drell, Albert C. Finn, Anthony C. Hearn.
Published in Phys. Rev. 136, B1439-51 (1964).

SLAC-PUB-42
HIGH ENERGY ELASTIC SCATTERING AT LARGE ANGLES AND THE STATISTICAL MODEL.
Ching-Hung Woo.
Published in Phys. Rev. 137, B449-51 (1965).

SLAC-PUB-43
ON THE 2pi DECAY OF THE K2-zero MESON.
John S. Bell, J. K. Perring.
Published in Phys. Rev. Lett. 13, 348-9 (1964).

SLAC-PUB-44
ON THE PROBLEM OF HIDDEN VARIABLES IN QUANTUM MECHANICS.
John S. Bell.
Published in Rev. Mod. Phys. 38, 447-52 (1966).

SLAC-PUB-45
HIGH ENERGY gamma RAY SOURCE FROM ELECTRON POSITRON PAIR ANNIHILATION.
Yung-Su Tsai.
Published in Phys. Rev. 137, B730-9 (1965).

SLAC-PUB-54
DEPOLARIZATION OF SPIN 1/2 PARTICLES BY ELECTROMAGNETIC SCATTERING.
C.K. Iddings, Gordon L. Shaw, Yung-Su Tsai.
Published in Phys. Rev. 135, B1388-97 (1964).

SLAC-PUB-59
THE INTERACTION EFFECT IN n-p CAPTURE.
H. Pierre Noyes.
Published in Nucl. Phys. 74, 508-32 (1965).

SLAC-PUB-60
THRESHOLD AND ASYMPTOTIC BEHAVIOR OF THE N/D EQUATIONS.
Myron Bander, Gordon L. Shaw.
Published in Ann. Phys. (N.Y.) 31, 506-24 (1965).

SLAC-PUB-61
BOOTSTRAP CALCULATION OF THE P MESON REGGE POLE.
Myron Bander, Gordon L. Shaw.
Published in Phys. Rev. 135, B267-7 (1964).

THEORETICAL PHYSICS (Cont.)

SLAC-PUB-62
DOUBLE POLES AND NON-EXPONENTIAL DECAYS.
John S. Bell, C. J. Goebel.
Published in Phys. Rev. 138, B1198-1201 (1965).

SLAC-PUB-63
DOUBLE CHARGE EXCHANGE SCATTERING OF PIONS FROM NUCLEI.
Ronald G. Parsons, James S. Trefil, S. D. Drell.
Published in Phys. Rev. 138, B847-50 (1965).

SLAC-PUB-64
THREE-NUCLEON PROBLEM WITH SEPARABLE POTENTIALS.
Myron Bander.
Published in Phys. Rev. 138, B322-5 (1965).

SLAC-PUB-65
PHOTOPRODUCTION OF NEUTRAL K-MESONS.
S.D. Drell, Maurice Jacob.
Published in Phys. Rev. 138, B1313-7 (1965).

SLAC-PUB-66
ON THE POSSIBILITY OF RELATING INTERNAL SYMMETRIES AND LORENTZ INVARIANCE.
Claude Itzykson.
Published in J. Math. Phys. 6, 1381-6 (1965).

SLAC-PUB-67
ANALYSES OF MUON ELECTRODYNAMIC TEST.
J. A. McClure, S. D. Drell.
Published in Nuovo Cim 37, 1638-46 (1965).

SLAC-PUB-69
THREE NUCLEON PROBLEM WITH SEPARABLE POTENTIALS.
Myron Bander. Nov 1964. 4p. Submitted to Phys. Rev. Lett.

SLAC-PUB-70
NONEQUIVALENCE OF THE ONE CHANNEL N/D EQUATIONS WITH INELASTIC UNITARITY AND THE MULTICHANNEL N/D EQUATIONS.
Myron Bander, Philip W. Coulter, Gordon L. Shaw.
Published in Phys. Rev. Lett 14, 270-2 (1965).

SLAC-PUB-73
SYTEMATICS OF ANGULAR POLARIZATION DISTRIBUTIONS IN THREE BODY DECAYS.
S. M. Berman, Maurice Jacob.
Published in Phys. Rev. 139, B1023-38 (1965).

SLAC-PUB-97
ABSORPTIVE CORRECTIONS AND FORM FACTORS IN THE PERIPHERAL MODEL.
Myron Bander, Gordon L. Shaw.
Published in Phys. Rev. 139, B956-61 (1965).

SLAC-PUB-102
ANOMALOUS MAGNETIC MOMENT OF THE ELECTRON, MUON, AND NUCLEON.
S. D. Drell, Heinz R. Pagels.
Published in Phys. Rev. 140, B397-407 (1965).

AUTHOR LIST

of

Reports, Preprints, and Reprints

February 1, 1969

Stanford Linear Accelerator Center
Stanford University
Stanford, California

Abarbanel, Henry D. I.
SLAC-PUB-371
THE STRUCTURE OF HIGH ENERGY LARGE MOMENTUM TRANSFER COLLISION PROCESSES.
H. D. I. Abarbanel, S.D. Drell, Frederick J. Gilman.
Published in Phys. Rev. Lett. 20, 280-3 (1968).

SLAC-PUB-390
THE CONSTRUCTION OF LOCAL QUANTUM FIELDS DESCRIBING MANY MASSES AND SPINS.
Henry D.I. Abarbanel, Y. Frishman.
Published in Phys. Rev. 171, 1442-52 (1968).

SLAC-PUB-411
LOW-ENERGY THEOREMS, DISPERSION RELATIONS AND SUPERCONVERGENCE SUM RULES FOR COMPTON SCATTERING.
Henry D.I. Abarbanel, Marvin L. Goldberger.
Published in Phys. Rev. 165, 1594-1609 (1968).

SLAC-PUB-476
THE STRUCTURE OF HIGH ENERGY PROTON-PROTON SCATTERING.
Henry D.I. Abarbanel, S.D. Drell, Frederick J. Gilman. Aug 1968. 44p.
Submitted to Phys. Rev.

Adler, Ronald J.
SLAC-PUB-33
MESON EXCHANGE EFFECTS IN ELASTIC e-D SCATTERING.
Ronald J. Adler, S. D. Drell.
Published in Phys. Rev. Lett. 13, 349-52 (1964).

Allen, M. A.
SLAC-78
RF CAVITY FOR SLAC STORAGE RING.
M. A. Allen, L. G. Karvonen, R. A. McConnell. Oct 1967. 35p.

SLAC-PUB-273
RF SYSTEM DESIGN FOR THE SLAC STORAGE RING.
M. A. Allen, R. A. McConnell.
Published in IEEE Trans. Nucl. Sci. NS-14, no. 3, 229-33 (1967).

SLAC-PUB-489
MATERIALS INVESTIGATION FOR A TWO-MILE SUPERCONDUCTING ACCELERATOR.
M.A. Allen, H.A. Hogg. Aug 1968. 12p.
Presented at Summer Study of Superconducting Devices and Accelerators at BNL, Jun 1968.

Allyn, Richard J.
SLAC-PUB-88
MICROWAVE AND FAST-ACTING VALVES AND VACUUM COUPLINGS FOR ACCELERATORS.
Arnold L. Eldredge, Albert J. Keicher, Marvin Heinz, Richard J. Allyn.
Published in IEEE Trans. Nucl. Sci. NS-12, no. 3, 694-8 (1965).

Altenmueller, Otto H.
SLAC-17
INVESTIGATION OF TRAVELING-WAVE SEPARATORS FOR THE STANFORD TWO-MILE LINEAR ACCELERATOR.
Otto H. Altenmueller, Rudolf R. Larsen. Aug 1963. 45p.

Altenmueller, Otto H. (Cont.)

SLAC-PUB-17

INVESTIGATIONS OF TRAVELING-WAVE SEPARATORS FOR THE STANFORD TWO-MILE LINEAR ACCELERATOR.
Otto H. Altenmueller, Rudolf R. Larsen, G. A. Loew.
Published in Rev. Sci. Instrum. 35, 438-42 (1964).

SLAC-PUB-135

DESIGN AND APPLICATIONS OF RF SEPARATOR STRUCTURES AT SLAC.
G. A. Loew, Otto H. Altenmueller.
Published in Proc. of the Int. Conf. on High Energy Accelerators, Frascati, Italy, 1965. Rome, CNEN, 1966. p. 551-5.

SLAC-PUB-224

BEAM BREAK-UP EXPERIMENTS AT SLAC.
Otto H. Altenmueller, E. V. Farinholt, Z. D. Farkas, W. B. Herrmannsfeldt, H. A. Hogg, Roland F. Koontz, C. J. Kruse, G. A. Loew, Roger H. Miller.
Published in Proc. of the Linear Accelerator Conf., Los Alamos, N. Mex., 1966. Los Alamos Sci. Lab. (LA-3609), 1966. p. 267-80.

Alvarez, Richard

SLAC-PUB-89

PRECISION PHASE ADJUSTMENT OF A LINEAR ACCELERATOR HIGH POWER WAVEGUIDE FEED NETWORK.
Richard Alvarez, R. Borghi, J. Weaver.
Published in IEEE Trans. Nucl. Sci. NS-12, no. 3, 39-43 (1965).

SLAC-PUB-181

ACCURATE PHASE LENGTH MEASUREMENTS OF LARGE MICROWAVE NETWORKS.
J. Weaver, Richard Alvarez.
Published in IEEE Trans. Microwave Theory Tech. MTT-14, 623-9 (1966).

Anderson, B. M.

SLAC-PUB-529

SLAC SPIRAL READER PROJECT.
B. M. Anderson, D. Budenaers, J. Caponera, R. Cochran, R. Good, C. Hoard, M. Hu, M. Jensen, H. Brafman. Nov 1968. 8p.
Presented at Int. Conf. on Advanced Data Processing for Bubble and Spark Chambers, Argonne National Lab., Oct 1968.

Anderson, L. R.

SLAC-56

A COMPUTER CODE FOR VARIABLE PERMEABILITY MAGNETOSTATIC FIELD PROBLEMS.
E.A. Burfine, L.R. Anderson, Habib Brechna. Apr 1966. 47p.

Anderson, M.

SLAC-PUB-270

PRECISION ALIGNMENT OF A LARGE BEAM TRANSPORT SYSTEM.
W. B. Herrmannsfeldt, M. Anderson, D. Connell, B. Hooley, J. G. Niforopulos, R. J. O'Keefe, E. J. Seppi, J. M. Voss, H. A. Weidner, J. K. Witthaus.
Published in IEEE Trans. Nucl. Sci. NS-14, no. 3, 903-7 (1967).

<u>Anderson, Robert L.</u>

SLAC-PUB-266
COHERENT PHOTOPRODUCTION OF THE eta-zero MESON FROM DEUTERIUM.
Robert L. Anderson, R. Prepost. Jan 1967. 16p.
To be Presented at the Int. Conf. on Low and Intermediate Energy Electromagnetic Interactions, Dubna, 1967.

SLAC-PUB-286
INTEGRATOR FOR NANOSECOND PULSES.
Robert L. Anderson, D. I. Porat.
Published in Rev. Sci. Instrum. 38, 1800-2 (1967).

SLAC-PUB-410
SLAC HIGH POWER HYDROGEN TARGET.
Robert L. Anderson. Sep 1968. 9p.

SLAC-PUB-413
SLAC HIGH POWER QUANTAMETER.
Robert . Anderson. May 1968. 12p. Submitted to Nucl. Instrum. Methods.

SLAC-PUB-428
HIGH ENERGY PHOTO-MESON PRODUCTION FROM HYDROGEN VIA THE 'u-CHANNEL'.
Robert L. Anderson, D. Gustavson, J. Johnson, D. Ritson, R. Weinstein, W.G. Jones, D. Kreinick.
Published in Phys. Rev. Lett. 21, 479-81 (1968).

SLAC-PUB-431
MEASUREMENTS OF pi-zero AND eta-zero PHOTOPRODUCTION AT INCIDENT GAMMA-RAY ENERGIES OF 5.0 - 17.8 GeV.
Robert L. Anderson, D. Gustavson, J. Johnson, D. Ritson, W.G. Jones, D. Kreinick, F. Murphy, R. Weinstein.
Published in Phys. Rev. Lett. 21, 384-6 (1968).

SLAC-PUB-439
FORWARD PHOTOPRODUCTION OF VECTOR MESONS FROM HYDROGEN AT ENERGIES FROM 6.5 GeV TO 17.8 GeV.
W.G. Jones, D. Kreinick, Robert L. Anderson, D. Gustavson, J. Johnson, D. Ritson, F. Murphy, M. Gettner, R. Weinstein.
Published in Phys. Rev. Lett. 21, 586-8 (1968).

SLAC-PUB-450
1.6 GeV/c CHARGED PARTICLE SPECTROMETER FACILITY AT THE STANFORD LINEAR ACCELERATOR CENTER.
Robert L. Anderson, D. Gustavson, R. Prepost, D. Ritson. Aug 1968. 19p.
Submitted to Nucl. Instrum. Methods.

SLAC-PUB-471
TIME-OF-FLIGHT SYSTEM FOR USE WITH THE SLAC 1.6 GeV/c SPECTROMETER FACILITY.
Robert L. Anderson, D.I. Porat. Sep 1968. 33p.

Yount, D. (Cont.)

SLAC-PUB-503

PHOTOPRODUCTION OF VECTOR MESONS WITH A 16 GeV BREMSSTRAHLUNG BEAM.
M. Davier, I. Derado, Darrell J. Drickey, D. E. C. Fries,
Robert F. Mozley, A. Odian, F. Villa, D. Yount. Sep 1968. 9p.

Zahn, C. T.

SLAC-70

TWO-DIMENSIONAL PATTERN DESCRIPTION AND RECOGNITION VIA CURVATUREPOINTS.
C.T. Zahn. Dec 1966. 109p.

Zarmi, Yair

SLAC-PUB-436

PROPERTIES OF THE P AND P' REGGE TRAJECTORIES FROM LOW ENERGY
pi N AND K N SCATTERING AMPLITUDES.
Frederick J. Gilman, Haim Harari, Yair Zarmi.
Published in Phys. Rev. Lett. 21, 323-6 (1968).

Zdanis, R.

SLAC-PUB-443

rho PRODUCTION FROM HYDROGEN BY 16 GeV BREMSSTRAHLUNG.
M. Davier, I. Derado, Darrell J. Drickey, D.E.C. Fries, Robert F. Mozley,
A. Odian, F. Villa, D. Yount, R. Zdanis.
Published in Phys. Rev. Lett. 21, 841-4 (1968).

Zipf, T. F.

SLAC-PUB-262

SECONDARY PARTICLE YIELDS AT 0-degrees FROM THE NEW STANFORD ELECTRON
ACCELERATOR.
Arpad Barna, Jack Cox, F. Martin, Martin L. Perl, T. H. Tan, W. T. Toner,
T. F. Zipf, E. H. Bellamy.
Published in Phys. Rev. Lett. 18, 360-2 (1967).

SLAC-PUB-324

ENERGY LOSS AND STRAGGLING OF HIGH ENERGY MUONS IN NaI(Tl).
E. H. Bellamy, Robert Hofstadter, William L. Lakin, Jack Cox,
Martin L. Perl, W. T. Toner, T. F. Zipf. Jul 1967. 13p.
Submitted to Phys. Rev.

SLAC-PUB-328

AN EXPERIMENTAL LIMIT ON HIGH ENERGY DIFFRACTION PHOTOPRODUCTION
OF THE phi MESON.
Yung-Su Tsai, Jack Cox, F. Martin, Martin L. Perl, T. H. Tan, W. T. Toner,
T. F. Zipf.
Published in Phys. Rev. Lett. 19, 915-7 (1967).

Zipf, T. F. (Cont.)

SLAC-PUB-434

A HIGH ENERGY SMALL PHASE-SPACE VOLUME MUON BEAM.
Jack Cox, F. Martin, Martin L. Perl, T.H. Tan, W.T. Toner, T.F. Zipf, William L. Lakin. Jun 1968. 34p. Submitted to Nucl. Instrum. Methods.

APPENDIX E: The System Development Process

The System Development Process

I. FOREWORD

The purpose of this narrative is to explain each phase of the system development effort and to enumerate the various activities that occur within each phase. Individual situations may call for deviations from this scheme, but the ideas expressed here are a general characterization of the events that will occur.

II. GENERAL OUTLINE (See Appendix F)

A system development effort can be divided into three general types of activity. The purpose of the first type is to define What the system is to do. The second type defines How the system is to do it. The third type consists of implementing the 'how'. These three types of activities are organized into six sequential phases. The 'what' activities are divided into two phases: the first is preliminary analysis and the second is detailed analysis. The 'how' activities are divided into general design and detailed design. The 'do it' activities are divided into implementation and installation. It is possible to place any systems development task into one of the six phases, and to state in which development function it belongs. Typically, these functions are Systems Analysis, User Requirements, Programming or a combination of the three.

III. PHASE CONTENT

A. Preliminary Analysis

Preliminary Analysis is that activity which must be undertaken prior to the detailed ennumeration of system requirements. First it is necessary in preliminary analysis to define the policies under which the development team will work, and to make a statement of goals that is both clear and general. The System Analysis team must then define the user environment in which the new system will operate. The user environment will include description of functions and activities within the user area, organizational structure and reporting relationships, the goals and charters under which the users work, and descriptions of the users themselves: their background, education, diversity of skills, etc.

The next effort which the system analysis team undertakes, is documentation of the current system, both manual and automated. Convenient vehicles for representing such a system are level one and level two flow diagrams which show every function and every sub-function within each user organization including representation of all document flow, between and within functions. Quantitative information is secured for each function and sub function, including throughput measurement in terms of numbers of discrete tasks performed per time span, number of documents per day crossing the interface between two functions, number of documents within a document receptacle, document turnover within a document receptacle, and similar data. Historical information of the same type is gathered wherever possible so that growth trends may be shown.

By identifying all such areas as a subregion of the overall user environment, it is possible to define a long term scope: those areas which could be improved if one were given all the time and all the money to do it. If the long term scope seems too large to cope with when treated as a single project effort, then it is necessary to define sub-scope alternatives for a first effort. These sub-scope alternatives are different combinations of those problem areas enumerated during the analysis of the current system. The analysis and selection of a sub scope out of this collection of alternatives involves choosing the set of problem elements whose solution would yield an optimum use of resources and present the highest return on the investment. A subordinate criterion would be the logical cohesion of the elements within a particular subscope.

After the sub-scope alternative for implementation has been chosen, it must be reviewed by senior level programming personnel from the project. At this stage, it is possible to render a gross judgment of the technical feasibility of the selected sub-scope alternative. It is possible that no judgment can be made at this point. In spite of this fact, however, misjudgments with respect to technical matters may be apparent. Given that they can be identified, the chosen sub-scope can be altered to reflect any needed corrections.

The Scope Document defines those portions of the user area to be focused on during the subsequent development process.

B. Detailed Analysis

The primary purpose of the detailed analysis phase is to enumerate in detail the requirements to be met by the new system. These requirements are divided into the following categories:

1. Performance

These are performance requirements stated quantitatively. Included are items such as response time, allowable mean failure time, maximum allowable down time interval, maximum allowable recovery time, hours per day of on-line accessibility, and maximum failures allowable in a given time span.

2. General Input-Output Requirements

Each input record should be given a descriptive name and have a proposed input medium. Estimates should be given of the total number of input records in a given time span, the growth of that number within a given time span, and a list of possible causes of fluctuation.
Any peak periods in the processing cycle may thus be isolated and provided for in the design. In addition, all those conditions causing generation of an input record by the user should be documented.

It is also important in the case of batch input to note any timing considerations associated with the processing of that input (Example: a particular report might be due in the user's area on the third working day of a month; The input feeding that report might not be available for processing until the second working day. Thus scheduling problems must be anticipated. in advance). The contents of the input record must next be listed. Each data element in the record must be shown, together with the criteria for editing in the data element and the action to be taken in case of rejection.

Output records must be defined in terms of a descriptive title, frequency of generation, suggested output media, sequence (batch only) and estimated volume per time span. A short paragraph must be written explaining how the information within the record will be utilized by the user. Criteria for generating the record must be shown (Example: this CRT matrix is generated in response to an on-line inquiry transaction.)
Finally, any timing considerations (batch only) connected with the publication of the output record should be noted. After defining an output record with these general parameters, the various data elements must be listed in the

output record. Each data element also requires a processing rule or a reference to a processing rule shown elsewhere in the Requirements Document. The Processing rule is used to obtain the data element. (See paragraph 4 below)

3. Detailed Design of I/O Documents

All input/output documents must be laid out character by character using forms appropriate to the I/O medium. Facsimiles of all documents to be used for key-punching must be produced. Formats for punched card input must be designed and documented keeping in mind both key-punching convenience and ease of processing. Punched cards used as output must be designed and documented with ease of interpretation in mind. Formats corresponding to typewriter terminal input transaction codes must be specified. Typewriter terminal output formats should be similiarly shown. Cathode Ray Tube outputs will be shown as a matrix on quadrille form. All printed report for mats will be individually laid out on the standard IBM printer spacing chart.

4. Transformation Rules

For every output data element there exists one or more corresponding input data elements. The correspondence between these data elements takes the form of transformation rules. These rules vary in complexity from algorithms and formulae to simple data transfer. The rules should be represented in a tabular fashion with a reference number corresponding to each rule. Decision tables may be used to represent individual rules wherever they apply. The reference numbers may be used elsewhere in the requirements document whenever reference to a particular transformation rule is desired.

5. Cost Limits

Cost limits are the upper bound of development and operating costs. They are derived by calculating the expected savings from the new system, plus the amount a user is willing to pay for services which do not now exist. If at any time during the development process, it is noted that the allowable cost will be exceeded, then a re-statement of requirements may be made to keep costs within the stated limit.

Items one through five above comprise an outline of the requirements document. This is the second primary development document, following the Scope Document. This draft is passed to the programming design team who analyze the technical feasibility of the enumerated requirements. Computer hardware needs are also examined at this time. The programming design team and the systems analysis team then meet in joint session to make any changes necessary to stay within the stated cost limits. At the completion of the detailed analysis phase the requirements document conforms to cost limits and technical feasibility.Project and user management then sign the approval page indicating the acceptance of the document as completed and accurate.

At this point the reqirements document is considered frozen. No subsequent additions or changes may be made to it unless approved by a control board composed of delegates from each group of signatories. The control board examines every change in terms of its cost, its impact upon the implementation cost, its impact upon the implementation schedule, and its complexity. They will decide if the change is to be included as part of the requirements or if it is to be deferred to a later iteration. In the latter case it will become part of the contents of the Project "Wishbook". This "Wishbook" will help to form the basis for preliminary analysis in a subsequent iteration.

C. General Design

The technical programming staff, with the Requirements Document in hand, will conceputalize alternative software solutions. The design team will then analyze each alternative using requirement satisfaction, development cost and operating cost as criteria. The alternative yielding the best combination of advantages will be chosen by a joint group of systems analysts,users, and programmers. The document corresponding to the chosen alternative is the General Design Document. The General Design is the third primary development document.

D. Detailed Design

Using the General Design Document, the programming staff will execute a detailed internal design extending down to the computer program module level. Corresponding to each module in the system will be a brief explanation of the purpose of the module, a detailed list of functions to be performed by the module (including needed algorithms, formulae, or coding systems), a list of interfaces (data elements to be passed to and from the module), programming language to be used, resource estimates in terms of man-hours and machine time to complete the programming of the module, and performance requirements. The systems analysis staff will identify procedures which must be added or changed. Training requirements and course outlines must be developed. Coordination with all affected user areas to insure complete understanding of the impact of the changes occurs during this phase.

This information,taken together with formats showing data element groups to be accessed or created by the module, comprise the programming specifications for that module. After the detailed design has been completed (programming specifications written for every module in the system), project management will create an implementation plan. This plan will encompass assignments and schedules for all project members for the duration of the development effort. A joint group of analysis, design and user personnel then create a testing plan which defines the manner in which the new system is to be tested. The scope of the plan will encompass unit testing, systems testing, and pilot testing.

The primary outputs of the detailed design phase are a complete set of programming specifications, an implementation plan, a training plan and a testing plan.

E. IMPLEMENTATION

Implementation is the portion of the development effort which gives practical effect and fulfillment to the design created in preceding phases. During this phase, the detailed training courses, lectures, etc. are completed. All required changes to manual procedures are completed and distributed. User manuals for both the manual and automated portions of the system are completed and distributed. Training courses for both the manual and on-line portions of the system are conducted, it is expected that virtually every member of the library staff will attend one or more training courses; some of the better trained staff will participate in the pilot testing of the system. This is described below. Each programmer on the project will be assigned a module or group of modules as tasks. In conformance with the specifications for these modules, he will create a set of computer instructions which, when executed, will yield the desired result. This endeavor is commonly referred to as coding.

Once a particular module has been coded it must be transcribed into a machine readable form either by the use of an IBM 2741 terminal or by producing a card deck on a keypunch machine. The machine readable version of the code must then be passed through a language translator (compiler or assembler) in order to transform the programmer's instructions into machine language instructions. Once this has been successfully accomplished, the programmer must undertake to test the coded module as specified in the testing plan.

Testing may be divided into three categories. The first category, Unit Testing, consists of executing the module either by itself or in combination with other modules that have previously been successfully tested. The next category, System Testing, implies an attempt to execute all system modules together and in the proper sequence. The data used for unit testing and systems testing are usually hand produced and specifically designed to test all paths of the logic. When meaningful output has been produced by Systems Testing, a joint group of project programmers and systems analysts will critique the results of the systems test. Programming modifications will be made as required.

The third category of testing, Pilot Testing, is similar to systems testing except that it utilizes live data in volumes characteristic of an operational situation. Pilot Testing is typically conducted by programmers, users, and systems analysts on the project working in close collaboration. After pilot Testing has produced results that are considered complete enough for evaluation, project personnel and users will critique the results. As before, modifications will be made to the system where needed to bring it into conformity with specifications. When all parties concerned are satisfied that the system is performing as it should, the Installation Phase is entered. It is necessary for the project members to prepare and submit a Systems Support Plan which will specify in detail how the project is to support the system after installation. The outputs of the implementation phase are tested programs; the results of unit, systems and pilot testing, are the maintenance documentation prepared in conformance with the project standards.

F. INSTALLATION

Installation is the process of changing from the old system to the new system. This involves converting manual files to machine readable files and reformatting where necessary to conform to the requirements of the new system. Once the data files have been transformed and are processable by the new system, all user procedures, hardware, computer programs, and user personnel begin operation. Certain portions of the new system may run for a period of time in parallel with the corresponding elements of the old system. In other cases, elements of the old system may not continue in operation beyond the cut-over point.

Old system elements running in parallel with the new system will discontinue operation only when all users are satisfied that the new system is performing satisfactorily. At this stage, the Support Plan prepared during the Implementation phase will become operative and certain members of the project team will be charged with the responsibility for performing maintenance and modification on the Production System. Project Management will undertake wrap-up operations consisting of finalization of the "Wishbook" (containing those extra facilities and extentions not included in the current implementation). Performance Statistics will be gathered on the new system running in a production environment for a 90 day period. Last of all a final narrative will be prepared under the supervision of the project director as a final addition to the project history. The output of the installation phase are processable data files, an Operational System, a "Wishbook", a set of performance statistics, an evaluation of those statistics, and a project history.

IV. NONPHASED ACTIVITIES

Nonphased activities are those ongoing efforts extending across phase boundaries which do not lie along the critical path of the project. All such activities are performed by the systems analysis function starting at the beginning of the General Design Phase. Project systems analysts will specify any needed organization changes within the user area and will generate procedures to be installed as required. Some manual procedures will be installed prior to production, other procedures will be installed with the advent of the new system. File build-up must begin in those areas involving retrospective conversion to prevent an interminable stretching out of the installation process. Installation of equipment such as terminals and display devices, also takes place. Each phase has major milestones which are either system events or document outputs. This is graphically represented in Appendices G and H.

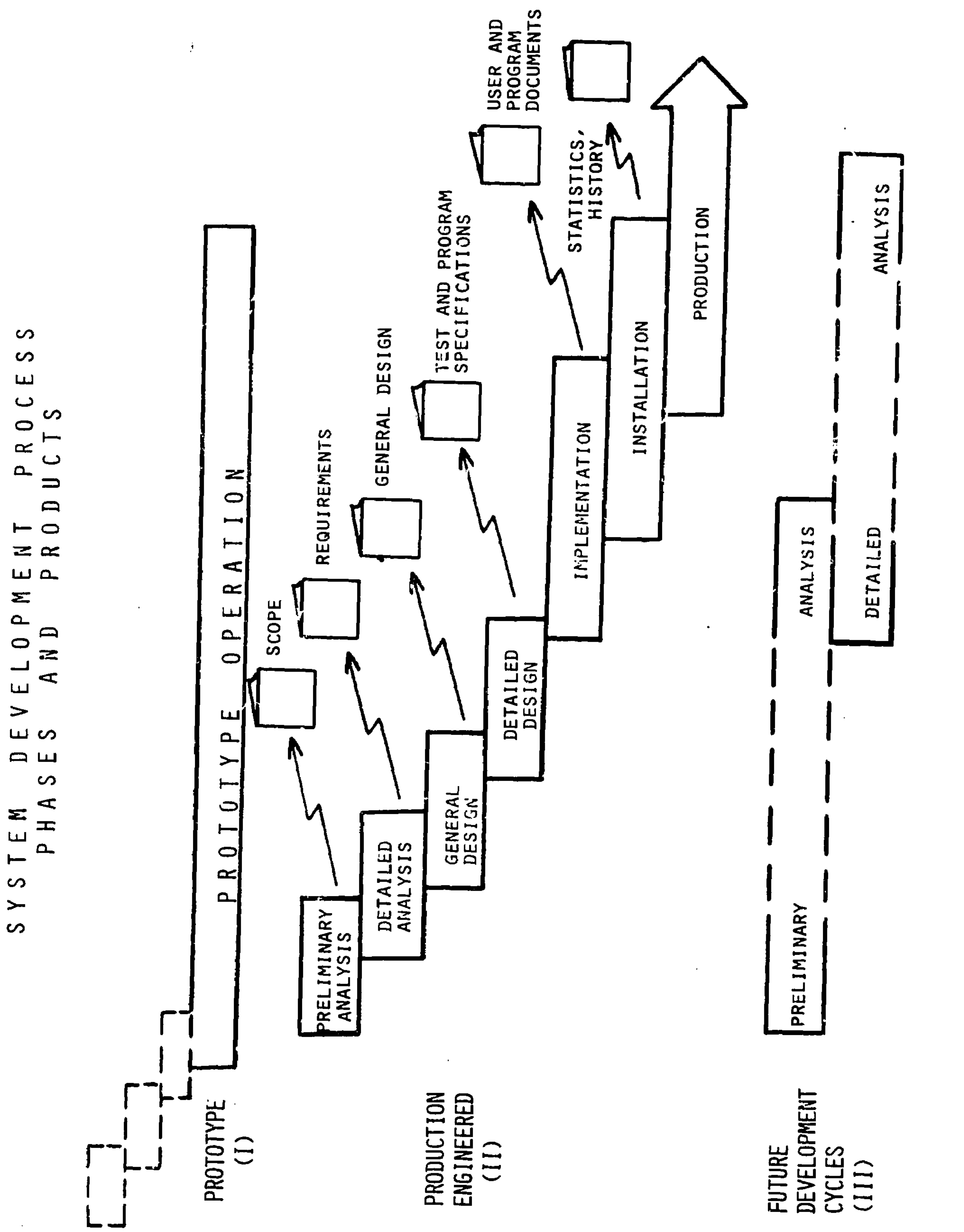
APPENDIX F: System Development Phase Activity

SYSTEM DEVELOPMENT PHASE ACTIVITIES

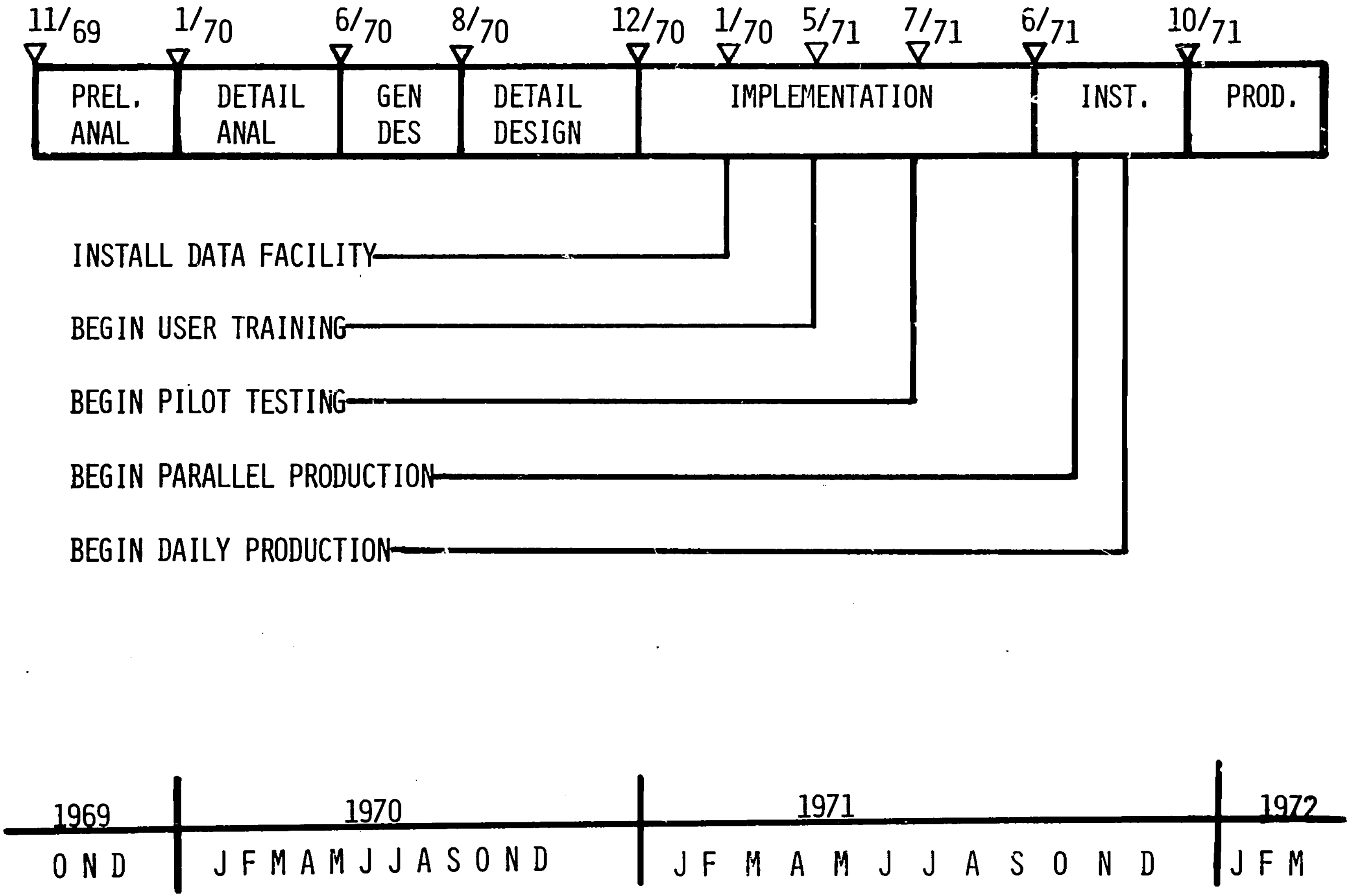
A -WHAT- B → C -HOW- D → E -DO IT- F

	PRELIMINARY ANALYSIS	DETAILED ANALYSIS	GENERAL DESIGN	DETAILED DESIGN	IMPLEMENTATION	INSTALLATION
SYSTEM ANALYSIS AND USERS	1) Define policy 2) Define goals 3) Define user environment 4) Document current system 5) Analyze current system 6) Define long term scope 7) Define alternative subscopes for first implementation 8) Analyze subscopes 9) Select subscope for first implementation	1) ENUMERATE REQUIREMENTS -Performance -General I/O -Transformation Rules -Volumes -Cost estimates and limits -Detailed design of I/O documents	0) EXTERNAL DESIGN Organization Procedures	0) EXTERNAL DESIGN -Training -File buildup -Related activities	0) EXTERNAL IMPLEMENTATION -Liason -Training of primary personnel -Equipment installation 0) Complete user and marketing documentation	0) TRAINING (secondary personnel)
PROGRAMMING, SYSTEM ANALYSIS AND USERS	10) Work out modifications	4) Work out changes where needed	4) Select best software solution	3) Create implementation plan 4) Create testing plan -- system test, pilot test	4) Critique system testing 5) Pilot testing 6) Critique pilot testing 8) Prepare support plan	2) CUTOVER -- parallel operation 3) WISHBOOK 5) Prepare final narrative
PROGRAMMING	1) Establish gross technical feasibility	2) Analyze technical feasibility of requirements 3) Analyze gross hardware needs	1) Conceptualize alternative software solutions 2) Conceptualize alternative hardware needs 3) Analyze alternatives 5) Choose hardware configuration	1) DETAILED INTERNAL DESIGN BY MODULE 2) Creation of programming specifications	1) CODE 2) UNIT TESTING 3) SYSTEMS TESTING 7) Modification as required	1) CONVERT FILES 4) Gather performance statistics
OUTPUT DOCUMENTS	SCOPE DOCUMENT	REQUIREMENTS *DOCUMENT*	GENERAL DESIGN *DOCUMENT*	PROGRAMMING *SPECIFICATIONS* IMPLEMENTATION PLAN. TEST PLAN	TESTED PROGRAMS USER DOCUMENTATION. MAINTENANCE DOCUMENTATION. TEST RESULTS	FILES; PERFORMANCE STATISTICS, SUPPORT PLAN, PROJECT HISTORY, WISHBOOK

APPENDIX G: Phase Relationships and Products



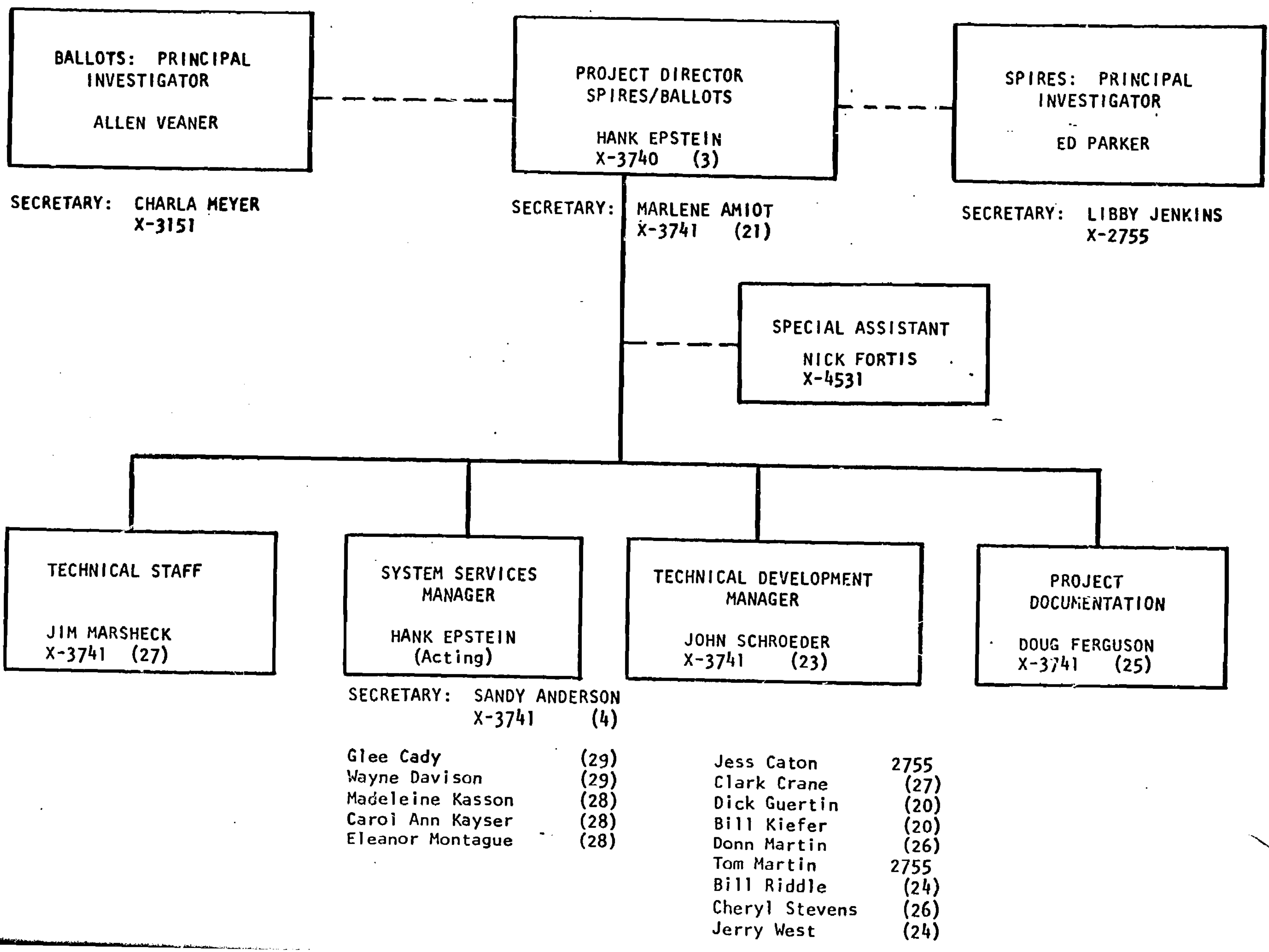
MAJOR MILESTONES AND SCHEDULES



JUNE 11, 1970

APPENDIX I: Project Organization

SPIRES/BALLOTS PROJECT



APPENDIX J: Scope of Spires II System (Excerpt from "System Scope for Library Automation and Generalized Informat on Storage and Retrieval at Stanford University")

7.0 CURRENT STATUS, GENERALIZED INFORMATION STORAGE AND RETRIEVAL

The SPIRES I Generalized Information Storage and Retrieval (GISR) Facility has been operating as a prototype System for approximately one year. During that time, the Stanford University Libraries, the Stanford Linear Accelerator Library, the ERIC Clearinghouse, the Department of History, and the Department of Geology have all built, maintained, and searched files on-line. Thus, it is seen that users of this facility do not fall into any particular organizational hierarchy, but are widely distributed geographically and with respect to academic discipline. Furthermore, the system now in existence and any system yet to be designed in no way changes the user organization or his procedures beyond those used for information gathering. These two facts make it necessary to weight the GISR discussion of current operations heavily toward software facilities as opposed to organizational divisions, functions, and processes.

7.1 Representative User Profiles

Various types of bibliographic users could easily make use of a GISR capability. There follows a brief sketch of seven possible user types. Refer to appendices E and F for detailed descriptions of law and physics users.

DEPARTMENTAL LIBRARIAN

Librarian Smith in a departmental library has been following the literature on machine-assisted bibliographic searching. A number of department members have made inquiries regarding a subscription service for computer tapes containing comprehensive bibliographic information in their field of interest. Librarian Smith does not know anything about computers but she is willing to learn in order to get a copy of the data collection. She does not do bibliographic searching for members of the department at the present time. I the future she would be willing to search the data collection for those professors who did not want to learn how to use the computer. Librarian Smith does not have any assistants.

RESEARCH LIBRARIAN

Librarian Brown of the university professional school library is an outstanding researcher. His library staff does most of the bibliographic searching for the faculty of the school, and occasionally for outsiders. He has determined that a considerable amount of searching time could be saved if the literature in an emerging field were properly indexed and kept up to date. He realizes that his school cannot afford to do this work in isolation, and so proposes to serve as a clearinghouse for indexing in the field. He is skeptical of computers but sees no manual method for preparing the material and keeping it updated without a large staff.

SENIOR RESEARCHER

Professor Black is a tenured member of the department and has an international reputation. He is a prolific writer and is the senior member of several research teams. Because of his heavy workload, he cannot afford to do bibliographic research personally. He hires graduate students to do the work, but is discouraged by the uneven quality of their work. If a device could be provided to allow him to search existing files exhaustively and rapidly, he could find what he needs more efficiently and use the graduate students for more exciting work.

EXPERIENCED RESEARCHER

Professor Lang has a collection of data relating to California. In his collection he has public opinion survey results, election returns, and census data. He wants to store this information on-line in card image format so that he and his students can test a series of behavioral hypotheses. Instead of listing the data resulting from a search (except for frequency counts, display of questionnaires, or candidate names) it would be saved for use by statistical routines.

INEXPERIENCED RESEARCHER

Instructor Jones is young and new to the department. He usually works alone because most of his colleagues do not work at the same pace. There is no adequate index to research literature in his specialty. Because of his experience with computers as a student, he wants to build a bibliographic data collection. He proceeds to build the collection and uses it extensively. After a year of work during which a 500 document collection is accumulated, his interest turns to a different problem in a related field. He moves to another university and his collection is abandoned.

RESEARCH ASSISTANT

Graduate student Johnson is a heavy user of the departmental library. He feels that he spends too much time trying to find material relevant to his interests. Since he has had experience with computers as an undergraduate, he considers it obvious that computers could be used to assist him. However he is afraid to rely too heavily on the computer since other universities might not provide the same services.

VISITING RESEARCHER

Mr. Peters is a graduate of the university but is now working in industry. He often needs to do research in his field. He feels uncomfortable when he visits the departmental library because he does not know anyone and does not know how the material is organized. He does not know much about computers and would use one only if led by the hand. He is willing to pay to get the help he needs.

7.2 Summary of User Requirements

The needs of the users profiled above form a wide spectrum. The requirements of Librarian Smith are complex and involve many capabilities for which library funds might be available; the graduate student has a well defined problem and at best a small budget to expend in solving it. Most other users fall somewhere between these two extremes.

ECONOMY & EFFICIENCY

The system must have a file structure that optimizes the trade-off between response time and disk storage utilization. Furthermore, the system software must be as efficient as possible while the hardware configuration must have just enough capability to do the job and no more. The cost for terminal time and for storage of information on-line must be low enough to be attractive.

SIMPLICITY

A successful system is usually simple to use. Some users have no computer background, and others have experience of relatively short duration. It is therefore necessary that a beginner be able to acquire the knowledge he needs with a minimum of research and study, preferably by having the system "lead him by the hand" during the initial phases. Furthermore, when the user commits an error, he should be directed toward resolution of his problem by a carefully conceived set of diagnostic messages.

FLEXIBILITY

The successful system must be user-adaptive, providing a variety of facilities to satisfy every need and pocketbook. A sophisticated system is obviously costly; if a simple and basic capability will suffice, the user should be given just that and charged accordingly. A consequence of this flexibility is that each user's file will look different. Thus, the need for AUTOMATED FILE DEFINITION (see 7.31.2 below) presents itself.

FEEDBACK

In order to evaluate the performance of the system, it is necessary to gather statistics which show the nature of the data stored in the system, the means used to retrieve it, and the frequency of access. Given such information, users may re-evaluate their file content and definitions in light of their experience, and make changes where appropriate. In addition, feedback must be provided regarding frequency of use (by user type and file type) and frequency of errors committed by users or by the system.

7.3 Summary of Current Facilities and Limitations

This summary of SPIRES I current facilities and limitations will entail brief descriptions of the two portions of the prototype system: data management and retrieval. Data management refers to the preparation, collection, formatting, storage, and maintenance of bibliographic information. Retrieval refers to the use of this information by people with the aid of the SPIRES/BALLOTS system. Both portions of the system are based on a file structure designed to provide maximum flexiblility in the placement and retrieval of data.

7.31 Data Management

Data management under SPIRES I refers to the manual-automated facility designed to handle data preparation, the establishing of files, file maintenance, and any special applications.

7.31.1 Data Preparation

The input of data into the system by local keyboarding and by conversion of data already in machine-readable form are the two means of data preparation. In either case, the end product is data in SPIRES Update Command Language format which is acceptable to the file building and updating program.

INPUT OF RAW DATA

The gathering of raw data is achieved by clerical workers using WYLBUR, the Stanford text editing facility. This method is more flexible for many applications than the alternative of keypunching card decks to be read into the system.

CONVERSION OF MACHINE-READABLE DATA

Large quantities of bibliographic data are available in machine-readable format. Such data is received on magnetic tapes which can easily be mailed from anywhere in the world. Conversion programs have been written to make some of these formats acceptable to the SPIRES system. DESY and NSA tapes (high-energy physics) can now be converted, as well as ERIC tapes (Education Research) and MARC (Library of Congress Machine-Readable Catalog).

SPIRES UPDATE COMMAND LANGUAGE FORMAT

The SPIRES Update Command Language format was designed for ease of encoding by human beings. It has, therefore, served its purpose adequately for data keyboarded locally. However, as a format into which to convert machine-readable data, the Update format has meant unnecessary inefficiency. A highly compact intermediate format into which to convert both SPIRES Update Command Language data, and other machine-readable formats is needed. Such an intermediate format would alleviate the decoding of highly compact machine-readable data into human-efficient format, which then has to be immediately re-encoded in the SPIRES files. Regardless of this drawback, the conversion process was a valuable feature of the SPIRES I system.

7.31.2 Establishment of Files

Prior to any file building or updating, files are defined and established. System programmers and users together determine how much disk space is required, the data elements to be used, data element values to be expected (format, length, multiplicity), which ones are to be indexed, and any special editing to be done. File definition under SPIRES I is done manually, and programmer assistance is required. An automated system was developed to interpret commands in a File Characteristics Language and generate a user-specific file definition, but it was not interfaced with the rest of the system. The next SPIRES system, in addition to automating the definition of these parameters, should look to other areas of user specification. For example, the definition of a large storage/low usage file might be distinguished from that of a small storage/high usage file, in such a way that efficiency and performance could be optimized in either case. This implies that the results of such file definition would be utilized by all parts of the system, not just by the data management portion.

7.31.3 File Maintenance

File maintenance under SPIRES I is accomplished by means of a batch mode record level Update facility. That is, one can add entries to the file and delete them, thereby replacing any entry. The use of storage in this task was geared toward reclamation of unused disk space. Therefore a dynamic file (heavily updated) would not grow indefinitely, but reach a point of space utilization equilibrium. In addition, statistics are kept regarding numbers of entries and data elements, and regarding questions of space and structure. Bibliographic entries are restricted in length to about 3500 characters of information and file size is limited by hardware capacity.

Various file management aids were developed to ease the task of the non-technical data manager. In particular, an experimental on-line macro facility was developed to aid the manager in such tasks as initiating build and update runs on the files, maintaining backup copies of those files on tape, and restoring files when necessary. This allowed the file manager to proceed somewhat independently from the system programmer in the file maintenance task. Further steps in this direction will be taken in future SPIRES/BALLOTS systems.

7.31.4 Special Applications

The development of any automated system involving files and useful information often encourages special applications not envisioned in the original system design. SPIRES I has been no exception. Data prepared for input to the system has also been used to produce PREPRINTS IN PARTICLES AND FIELDS, a weekly newsletter containing the most important bibliographic information sorted by key. In addition, the SPIRES data base has been used to produce for SLAC a semiannual publication containing bibliographic descriptions of articles by local authors only, sorted by author, subject, and key.

7.32 Retrieval Facility

The process by which bibliographic data is entered into the system and kept current has been discussed. What follows is an explanation of the means by which data is retrieved.

The SPIRES Retrieval system is a fully automated on-line bibliographic search capability allowing the remote terminal user to make various search and output requests.

7.32.1 Search Requests

Once communication is established with the retrieval facility, the SPIRES user must select a specific file for bibliographic searching. For example, he might choose the SLAC Preprint file or the Geology file. The user may then begin an interactive search session on his selected file. Depending on his choice, he may search on such indexes as are available for that file. Author indexes can be searched on names in a variety of conventional formats (first last; last, first; etc.). Titles are searched by specification of one or more title words or title word stems which do not appear on the system exclusion list (words too heavily used to be meaningful as search items). Citations require a more rigid format: journal description, volume number, page number. The user may interactively narrow or broaden his search by compound search requests, using the connectives AND, OR, and NOT to combine search terms from any index. Search results may be further narrowed by specification of dates: BEFORE, AFTER, FROM, SINCE, or THRU may be used. If the searcher finds he has inadvertently narrowed his results too far, he may BACKUP to his earlier findings.

7.32.2 Output Requests

At any point in the search session, the user may interrupt his searching and have his accumulated results typed at his terminal. He may use the standard SPIRES output format, which includes all data elements in each document and their associated values. Or, he may select certain data elements to be listed in a specific order. In using this second option, the user could have the title printed first, and if it were of interest to him, allow the rest of the document description to be printed out, otherwise interrupting the output and going on to type the next entry.

7.32.3 General Comments

A SPIRES Reference Manual has been published which contains a step by step description on the use of the SPIRES Retrieval Facility. It would have been desirable to have incorporated more of this training into the system itself in order to ease the user-initiation process. This would imply a more extensive error diagnostic and error recovery capability. In terms of output of search requests, a print off-line capability is certainly needed. Another feature needed in a future version of SPIRES/BALLOTS is the manual and automated use of statistics on the retrieval facility to improve overall system performance, efficiency, and responsiveness to users.

8.0 Long Range Scope for Generalized Information Storage and Retrieval

The preceding section dealt with the present system, SPIRES I. This section defines those facilities to be eventually added to the system. It must be noted that some, but not all, will be chosen as a Scope for Implementation in the next development iteration.

8.1 Retrieval

Retrieval requests will have two essential parts: a search request and an output request. A series of iterative search requests, each giving feedback to allow framing of subsequent requests, will state the criteria which the user wishes any retrieved record to meet. The output request will state which data elements of the retrieved records he wishes to see. These facilities will be available for both on-line and batch operations.

8.11 Search Requests

INDEX TYPES

The basis for on-line retrieval is the set of indexes associated with a file. There exist many kinds of indexes; each index represents a different way to enter the file. Some examples are given below.

1. Personal name indexes: Personal names consist of alphanumeric characters. Names are indexed with surname first, followed by given names (or initials), followed by title, if any. For example, the name "Sir John Gielgud" would be indexed as "Gielgud,John,Sir". In retrieval, this allows matching on phonetic representations, surnames only, surnames and initials, or an exact match on the full name, e.g., FIND EMPLOYEE MOOK, EMPLOYEE MOEK, or EMPLOYEE L. MOEK, or EMPLOYEE LARRY J. MOEK. The more specific the request for a match, the fewer matches are found.

2. Title word index: Titles consist of one or more words comprised of alphanumeric characters. Each significant word in the title phrase is indexed separately. In retrieval, a match on a single word will retrieve all titles containing that word. A match on a word phrase could result in retrieving all titles containing all the words in the phrase regardless of order, e.g., FIND TITLE HONEY BADGER would retrieve the titles: THE HONEY BADGER and THE BADGER WHO LIKED HONEY. Alternatively, specification of a

word phrase could result in retrieving titles containing an exact match, e.g. FIND EXACT TITLE HONEY BADGER would retrieve only the title: THE HONEY BADGER.

3. Topic index: Topics, keywords or subjects are all synonymous with the concept of specifying words and phrases which describe the subject matter treated in a document. Topics consist of one or more words comprised of alphanumeric characters. The entire phrase is indexed as a whole, not separated into individual words as with titles. In retrieval, the exact word or phrase is matched with order preserved.

4. Numerical indexes: Numerical indexes contain data element values comprised of integer characters. Each data element value is indexed once, e.g., numbers assigned to parts in a garage supply warehouse. Another type of numeric index would enable users to retrieve from a range of numeric values rather than only one specific value.

5. Date indexes: Since dates may be entered in various formats, they will be converted to a standard format before they are indexed. Examples of dates are: DATE OF PUBLICATION, DATE ADDED TO FILE, etc.

6. Coded indexes: Codes are comprised of alphanumeric characters. The code value is indexed once and matches for retrieval are made on the complete value. Dictionaries are used to convert the codes to their full equivalent. An example is a large manufacturing concern with many outlets across the country. Each outlet is assigned a code so as not to maintain the full name of the outlet in the indexes.

7. Broad classification on indexes: Some document collections can be broken into a few broad classes. When it is desired to index that kind of data, special consideration must be given to the fact that all the data falls into just a few groups. An example can be drawn from the SLAC Preprint files where all documents can be classified as containing experimental, theoretical or instrumentation information. It is desirable to be able to access the files of data through this classification, e.g.,all documents by Jones in experimental physics.

The above examples do not comprise an exhaustive list. Most data elements to be indexed can be classified into one of these categories. Facilities will nonetheless exist for defining those that do not.

MULTIPLE LEVEL ACCESS

In addition to the ability to define multiple access points for a file, users will have the ability to divide a file into several levels. Indexed elements will be used to select a set of records from a file. This set may be further searched using set of indexed elements or may be searched sequentially to check non-indexed elements against another set of criteria. For example, a search might be performed on a set of insurance policy files for all policies of a particular type issued during the year for a face amount of $5,000 or more. In this example, the access points would be the policy type and date. The sequential search would be performed on the amount.

BATCH SEARCH

An alternative to on-line retrieval will be batch retrieval. Batch requests may be formatted on-line, and syntax checked for correctness of structure. They will then be accumulated for later processing against the desired file. The file will be searched sequentially for matches of requests with stored information. To minimize repeated passes over the same items, the requests may be grouped so as to find all requested information from the first record before moving on to the next.

Batch retrieval restricts the way one formulates a search request. A user will not have the ability to expand or contract a set of selected items resulting from a single batch search. Several more batch searches may be required before the user finally retrieves the desired set of documents. In contrast, the manner in which one formulates a query for on-line retrieval of information is dependent upon the ability to access that information directly without passing over previously stored information. One can skip back and forth within the file gathering information, expanding or contracting the set of selected items, and examine the contents of that set when desired -- all during one session at the terminal.

SIMPLE SEARCH REQUESTS

In stating a query, the user will indicate which element or elements he wishes to access, e.g., AUTHOR. He will then supply a value against which all values for that particular element are compared, e.g., AUTHOR JOHN BROWN. Such a query would be a "simple request".

COMPOUND SEARCH REQUESTS

A facility will be available to construct compound requests. Simple requests may be combined into a logical expression by using the words "and", "or" and "not". The use of "and" will allow the user to specify two or more criteria which all the records retrieved must satisfy, e.g., AUTHOR BROWN AND TOPIC NUCLEAR PHYSICS. Using "or" will allow the user to specify two or more criteria, at least one of which must be satisfied in each record retrieved, e.g., AUTHOR BROWN OR AUTHOR JONES. The use of "not" will allow the user to specify a term which is to be excluded from the set of retrieved records, e.g., AUTHOR BROWN AND NOT AUTHOR JONES.

In addition to the logical expression capability, one will be able to group simple or compound requests so as to imply logical preference or ordering, e.g., (AUTHOR BROWN OR AUTHOR JONES) AND TOPIC NUCLEAR PHYSICS. In this example, parentheses are used to indicate a preferred grouping. Everything within the parentheses would be evaluated prior to performing the remainder of the request. One would be able to nest these groupings as in (AUTHOR BROWN AND ((AUTHOR JONES OR AUTHOR SMITH))) AND TOPIC NUCLEAR PHYSICS.

In response to a request, the system will indicate to the user the number of items found in the specified file for each simple request. If the request was formulated as a logical expression, the system will respond with the number of records that satisfy the complete request. The user now has several options. He may choose to browse through the content of the records, i.e., make a request of the output facility described later in this section. He may choose to begin a new search request on the same file or on another file. Or, he may wish to modify the previous request. By modifying the request, the user would expand or contract the set of retrieved records. For example, the request:

FIND AUTHOR JONES OR AUTHOR BROWN

might retrieve 75 records which have either JONES or BROWN as an author. The user might then enter:

AND TOPIC NUCLEAR PHYSICS

which will reduce the set to those documents which have NUCLEAR PHYSICS specified as a topic. The user may find he has narrowed his search too far and may then choose to use an OR to expand the set. If at any time the user finds he has made a poor choice of criteria, he will be able to return to some previous point in his query and start again from that point.

A search request may be qualified with a date. A search may be limited to only those items before or after a specific date or within a range of dates. This facility will allow a user to search through current information, i.e., that portion of a file added since some date. Other dates that could be used in this way are publication date, date added to file, etc.

WEIGHTED SEARCH REQUESTS

The search facility, as it has been described so far, is a "hit or miss" process. Either all criteria are satisfied for a specific record or nothing is retrieved. One may therefore wish to attach percentages or weights to the search terms in a request. Through the use of this facility, he will specify that all items he found which contain a specific number of a given set of terms, e.g., find all documents which contain any three out of five given terms. Another way of attaching weights to particular terms would be to submit a request for all records found exceeding a specified score, when each term is assigned a weight. For example, the following request:

FIND TITLES (METAPHYSICS, 7 EPISTEMOLOGY, 5 ONTOLOGY, 5 EXISTENCE, 4 PHILOSOPHY, 3) WITH TOTAL SCORE 9

states that all documents are to be found having titles with a combination of the words in parentheses, such that the sum of the attached numbers is nine or greater. Thus, the bibliographic items for the titles "Epistemology as a Philosophical system" and "Epistemology and Ontology" would be retrieved, whereas those with title "Existence - a Philosophical Examination" or "a Philosophical Examination of History" would not. This facility is generally called weighted searching.

An alternative scheme would provide for the specification of weights in terms of decimal numbers less than one, with search results ordered by descending score.

CORRELATION OF SEARCH REQUESTS TO ABSTRACTS

If a bibliographic file had a data element which contained abstracts, a retrieval criterion could be stated in terms of one or more English sentences. The retrieval process would correlate the given phrase with each abstract and retrieve those records containing abstracts with a correlation coefficient greater than some specified value.

It should be noted that Salton et al. at Cornell University have been experimenting with this facility for some time, but have not implemented an economical system. Such a facility lies beyond the current economic boundaries for SPIRES II.

DICTIONARIES

Dictionaries will be available to assist the user in selecting search terms. Some dictionaries may be general and applicable to all files while others may be specific to a particular set of related files. Dictionaries containing exclusion words, synonyms, codes and abbreviations would be specific to a set of related files. Dictionaries of this type will be built at the time a file is established and relate to the content material. The user will have the option to modify basic lists provided by the system to meet his own requirements.

A user may use synonym, and abbreviation dictionaries to guide him toward a selection of terms which are appropriate for the particular file from which he is retrieving information. A file may contain abbreviations unfamiliar to the user. He may be using a meaningful word or phrase in his request, but the file manager preferred and used another word or phrase in his indexing.

Some information may be stored in a file in coded form to conserve space. A dictionary is needed to find the full equivalent which the codes represent, e.g., scientific journal names maintained as coded data in the file with a dictionary giving the full names of the journals and their associated codes.

For other elements of a file, there are values or words which either have no significance as far as content is concerned or occur too frequently to be of much value in retrieval. For such elements, a file manager may construct a dictionary called an exclusion word list. Words on this list would be dropped from any request which included them. The user will have the facility to interrogate these lists.

THESAURUS FACILITY

The thesaurus facility will be closely related to dictionaries. A thesaurus is file-specific and may contain a list of synonyms for key words or phrases used in a file. Reference to this list will enable the user to select other words and phrases which would assist him in retrieving additional relevant records. A thesaurus may also show hierarchical relationships among words. The user will be able to reference this list to find those words or phrases which are related to the same topic but are more specific or more general in nature. A thesaurus could be constructed and access to it provided for the user to determine the general nature of topics covered in that file and, thus, serve as a "jumping-off-place" for his search.

INDEX REFERENCE

The user will have the capability to list indexes and use the results to formulate more accurate search requests. Also provided will be an item count corresponding to each index term.

TRUNCATION OF SEARCH TERMS

Another facility which will be helpful to the user at the time of formulating his request is the ability to truncate search terms. This facility will enable him to use words without suffixes, thus retrieving records from a file in which various forms of the word are contained. For example, in the request:

FIND TITLE WORK#

the '#' symbol has been used to signify truncation. Assuming the TITLE data element had been indexed for the file being accessed, the records with titles containing the words WORKS, WORKING and WORKED would be retrieved. Truncation also may be used where the spelling of a term is doubtful as:

EMPLOYEE HAN#

Employee records with surnames HANLEY and HANDLEY would be retrieved. The user may then be more specific once he has determined which record satisfies his interest.

A facility similar to truncation will provide for alternative spellings. A search term would be specified with 'don't care' indicators, as in the example below:

EMPLOYEE HANS#N

The ambiguous '#' would cause employee records with surnames HANSEN and HANSON to be retrieved. This would be useful in cases where the exact spelling is unknown. It would be necessary, however to specify at least the first three letters of the name before inserting 'don't care' characters. Truncation options will be provided for searching name, title word, and topic indexes.

SAVE-REUSE FACILITY

A save and re-use facility will be available. At any point within his search request, the user may save the results of his query for later use. He may also save and re-use the request itself.

STANDING REQUESTS

Users may be only interested in any new information which has been added to a file. The standing request facility will be helpful here. Users need only formulate their requests once and leave them with the system. Information which is being added to a file will be passed against the requests and any matching records delivered to the requester.

RECOVERY OF SEARCH RESULTS

If something happens within the system causing interruption of normal service, users should be restored to their place in the search. This should be the responsibility of the system and not the users.

8.12 Output Requests

GENERAL

SPIRES will accept output requests which allow selection within the following options: media, format, document selection, sorting, and generalized report format/content.

OUTPUT MEDIA

The system will provide a spectrum of output media from which a user may choose one or several - appropriate in terms of cost, output volume, convenience (usability), and reusability (machine-readability).

If his output volume is low, the on-line user may be satisfied to accept it from the terminal communication devices: typewriter or CRT. The typewriter supplies him with a hard copy whereas the CRT does not. Since the typewriter is relatively slow and only one line may be listed at a time, flexibility provided via this device will be minimal. The CRT can display several lines at a time, thus providing better formatting and giving the user a scanning facility. The capabilities of the CRT will allow the user to browse through a set of selected records at his own pace.

If his output volume is high or he desires a permanent copy, he can divert it to an off-line batch process: to either a high-speed line printer or computer output microfilmer (COM). The printer output format can be varied in the forms or print chain used, and the number of copies prepared. The microfilm option has three advantages over the printer option: the microfilm requires little storage space, it can be searched and viewed manually or mechanically, and it can be used to produce unlimited hard copy at a small percentage of the cost.

Finally, if his output data must be re-read by the computer at a later time, he can choose magnetic tape, magnetic disk, or punched cards as his output medium. Information stored in this way can also be subsequently listed or distributed externally, e.g. sending a tape to another institution.

OUTPUT FORMATS

Information may be presented in various forms. The user will have a choice in the data elements in each record he wants to see and the sequence in which those elements are to be presented. If he creates a format which he will want to use at another session, he will be able to save the specification and re-use it later.

There are three sources of formats:

1. System-wide standard
2. File standard
3. User-defined

All three sources will be available to the user. He will be notified if he has used a format which is inappropriate for the file in which he is currently working.

The user will be able to set tabs at his typewriter terminal to affect column assignments and margins, set a line length to limit the number of characters to be presented on a line, set a page or screen length for number of lines, and request labels attached to the elements presented. The formatting features provided at the terminal will be limited and straightforward because of the excessive time required to produce sophisticated output on-line.

SELECTING DOCUMENTS FOR OUTPUT

At the time a user asks for the contents of selected records to be listed at his terminal, he will be able to:

1. Specify a range of records or a selection of records to be presented, for example:

   TYPE 1-5,10,15

   where only those items indicated would be presented, skipping the rest of the set;

2. Ask for all records in sequence beginning with the first;

3. Ask to be given an option after viewing each record, which permits its storage for later use.

4. Interrupt the listing at any time and,

   a. resume with the interrupted line,
   b. skip to the next record,
   c. skip to a specific record,
   d. skip to the end,
   e. leave the output process entirely,
   f. leave the process temporarily, and return later.

OUTPUT SORTING

Another process concerning presentation is the facility for sorting on one or more data elements. For example, personnel records may be sorted alphabetically by employees' surnames.

DECODING DATA ELEMENTS FOR OUTPUT

If data elements have been stored in coded form, the user has a choice of seeing the information in its compact or expanded form.

REPORT GENERATOR CAPABILITY

A report generator will be provided to allow the user to produce batch listings of selected data base elements in formats of his design.

8.2 FILE MANAGEMENT

8.21 General

There are several needs to be filled in the area of file management. The first of these is a facility for a file manager to define the characteristics of his file without requiring the aid of a programmer. He should be able to enter the specification of the characteristics through a terminal in a non-technical language. Further, the file definition facility must give as much aid as possible in diagnosing errors in the specification. It also must have the capability of allowing the manager to make reasonable alterations to characteristics after the file has been built without having to completely re-build the file.

8.22 Establishment of Files

STORAGE SPECIFICATION

The first characteristic to be specified by the file manager is the amount of direct access storage that will be required for the file. This estimate will be based on the amount of data to be entered in the initial buildup of the file, the rate of growth, and the indexes chosen. The initial allocation of storage should be sufficient to hold the initial data plus the additions which will accumulate over a period of several months. The system must be able to extend the storage for any given file either automatically when the previous allotment has been exhausted or on the entry of a simple command by the manager. If the latter alternative is implemented, the system should issue a warning when the data in the file approaches the current storage capacity.

SPECIFICATION OF DATA ELEMENT ATTRIBUTES

The file manager must decide how to separate the documents into data elements and specify their properties. The properties to be specified are: element name, abbreviations and synonyms, multiplicity, element size, data type, editing functions, and any hierarchical relationship to other elements. The name of an element may contain any of the characters on a terminal keyboard but, for retrieval purposes, an abbreviation must be specified. If it is not, the system will create one. The element size is the number of characters contained in a fixed-length element or the maximum number of characters for a variable-length element. The system must support the following data types: numeric, data, personal name, alphabetic, coded, and full text. Other data types which might be supported are: monetary, linear measures, weights, fractions, and sets of related numbers. Standard sets of data element characteristics may be maintained by the system. Thus, a file manager may elect a default of one or all of these, if it applies to his file.

HIERARCHICAL RELATION BETWEEN DATA ELEMENTS

The concept of a hierarchical relation can best be described with an example. Suppose a file was established with each record being the description of a piece of electronic equipment. Each piece of equipment might be composed of a set of components. One data element might contain the identifications for each of the components. Associated with each value of the component ID element would be an element containing a list of part numbers for that component. Associated with each part there would be an element containing the price of the part. Another example to illustrate hierarchical relations is provided by a bibliographic file where each record represents the reference material for the preprint of a scientific paper. One data element would contain a list of authors of the paper. For each author, one element would contain the organizations with which he is affiliated and an associated element would indicate his mailing address at that institution. Still another element might contain his title with that institution. The system should not place an arbitrary limit on the number of these relationships that may exist among the data elements of any given file.

SPECIFICATION OF INDEXES

Since the manner and degree in which the file is indexed is vital to the retrieval capability that the users will have in accessing that file, the facility given the manager for tailoring the indexing to the requirements of his particular file is extremely important. He should be allowed to specify indexing for any combination of data elements and to have values of more than one element entered into a single index. In addition, it should be possible to add a new index or delete an existing one after the file has been built.

SPECIFICATION OF EDITING RULES

The manager must have the capability to specify editing for the values of an element to be placed in a principal file. Normally, this consists of making selections from a standard set of editing procedures, e.g., function words (like THE, OR, BUT) may be excluded from an index. The manager should also be allowed to specify special editing procedures although he may be required to pay for any programming costs associated with them.

An additional facility would require that the presence (or absence) of a value for one data element necessitates the presence (or absence) of a value for some other element.

DICTIONARY/THESAURUS SPECIFICATION

The File Manager will have the capability to define dictionaries which are specific to a particular set of files. The definition will be part of the file characteristics placed in the system by the File Manager preceding the initial file buildup. A similar capability will exist for Thesauri.

FORMAT SPECIFICATION

Since a user retrieving from the file should not have to specify the format in which information will be displayed on his terminal, some facility is required for assigning standard formats to the file. These formats may be selected from a set provided by the system or the file manager may define some to meet specific requirements of his file.

8.23 File Maintenance

UPDATE

It should be possible to carry out the update function in any of three ways: completely on-line, completely on a batch basis or as a combination of the two. In the on-line mode, update requests would be entered via a terminal, immediately checked for errors and the change in the file executed while the user is still at the terminal. Batch updates could be punched on cards and delivered to a computer operator who would then place them into the batch queue. An intermediate alternative would allow the user to enter the requests from a terminal but allow the system to collect them into a batch and place them in a queue for later execution. In all cases, the system will have a facility to list the updates that were executed for the file.

Three categories of update requests are needed. The first is the addition and deletion of records. The second is the addition, deletion and altering of data elements within records. The third is to be able to copy information from one record to another or from one file to another.

In order for a user to be able to specify, with ease and without fear of ambiguity, which record of the file is to be updated, it is necessary to have a data element which contains a unique value. This data element must be indexed and the system should check each entry in that index to insure that it references only one record in the file. Examples of this kind of data element are; social security numer in a personnel file, Library of Congress card number, and part number in a parts inventory file.

8.24 Output for File Managers

In addition to the output needs for the support of the retrieval function, two special outputs will be required by some file managers. The first of these, to be used to augment the on-line services or to disseminate externally, consists of catalog cards or shelf lists. These must be sortable on at least one data element.

The second output will consist of various statistical descriptions of the file as prescribed by the manager and gathere by the system. These statistics will aid him in predicting the growth of the file, in determining the utility of an index and in various other management tasks. Examples of statistics he might need are; average lengths of data elements, number of times an index is used in single or cumulative retrieval requests, or quantity of each kind of error made in update requests.

8.25 Training

Several facilities will be needed for training file managers and persons who will be assisting them in maintaining files. A consulting service will be necessary for the dual purpose of aiding the managers to establish their files and helping the file maintenance people when they have difficulties with the update function. In addition, classes should be given from time to time to introduce newcomers to the capabilities of the system.

Several kinds of reference material should be written and made available. These are: a primer, a complete file management reference manual, a short version of the reference manual for maintenance people, and reference cards. The last would be very brief excerpts from the manual printed on cards. They would serve principally as reminders to users while on the terminal.

Once a user is communicating with the system, various online aids should be available. He should be able to ask for a brief introduction to the facilities for file management, for examples of the use of these facilities, for explanation of particular terms and prompts, and for an explanation of what responses are available to him.

8.26 Individuation of Retrieval

In the sections above, much attention has been given to facilities to enable a file manager to individualize his file and tailor it to suit his information and retrieval requirements. In addition, it would be useful for the system to provide certain facilities for individualizing the retrieval function to the habits and idiosyncrasies of a particular searcher. The facilities which might be implemented for this purpose are: macros, subset indexes, subset language, unobtrusive observation, and service priority.

SEARCH MACROS

In the macro facility, the user would be able to combine several requests into one and assign a name to it. Subsequently, he could cause the set of requests to execute by entering the macro name. This feature would reduce the effort of users who repeatedly carry out some particular sequence of requests. For example, suppose someone frequently entered some term of a topic index, requested all synonyms, assembled these into a retrieval request for all records containing any one of them and finally requested to look at the first three of the retrieved records. If this were all combined into a macro, it would save him a significant amount of keying and possibly some mistakes.

SUBSET INDEXES

At times, some user might wish to do exhaustive searching though part of a very large file. For example, a geologist might wish to work with the section of an earth sciences bibliographic file which pertains to precious metals. In order to reduce the cost of the on-line retrieval, it would be advantageous for him to be able to request the creation of a file which would be a subset of the full earth sciences file. To achieve minimal cost, the file records themselves would not be duplicated, but rather, a separate set of smaller indexes would be built.

LANGUAGE SUBSETS

In order to make the process of entering retrieval requests simpler and thus reduce both the amount of learning required and the number of errors made, the system should support language subsets. A user would only need to learn those request formats which apply to his individual needs.

UNOBTRUSIVE OBSERVATION OF USER HABITS

Most users will probably make certain errors quite frequently. If a record were maintained by the system of each user's habits, then, for those errors which are made consistently (and also corrected each time by the user), the system could make the correction for the user. This facility should, however, be an optional one.

USER PRIORITY

Normally the system will consider the requests of all users to be of equal importance and will optimize the servicing of requests to keep the average response time to a minimum. However, on some occasions, a particular user may have need for faster service and be willing to pay for it. Thus the system should provide the facility for a user to assign priority for his requests and to charge him higher rates accordingly.

9.0 Generalized Search and Retrieval First Implementation Scope

In order to fully understand this section, it is necessary to have read Chapter 8.

The system will have the following general characteristics:

1. Flexibility - the system must be able to accommodate a variety of files, including any of the bibliographic data available in machine-readable form.

2. Adaptability - it must be possible for a user to use and be charged for only that part of the system which he needs.

3. Modifiability - the system should be designed and implemented in such a way that it is easy to change. In particular, it is foreseen that the interactive search may require expansion.

9.1 Retrieval

The following search facilities will be implemented:

1. indexes - the user will be able to use indexes of the following types in his search requests. However, for any given file, he may use only the indexes associated with that file.
   a. personal name
   b. title word
   c. topic - contains terms descriptive of the subject matter of documents in a bibliographic file.
   d. numerical
   e. date
   f. coded
   g. file partition - ability to divide a file into sections. For instance, a file of physics papers might be partitioned into experimental, theoretical and survey sections.
   h. user-defined - the data type, editing or format is specified by a file manager especially for his file. If any additional implementation cost is

required, it will be at his expense.

1. citation
2. access via non-indexed elements
3. on-line search
4. batch search
5. query language
   a. logical expression - several simple requests may be combined into one request by use of the words: AND, OR, NOT. For example: FIND AUTHOR Smith AND TITLE Hemophilia.

   b. weighted terms- each term of a request may be assigned a number by the user. Only those records which score the same or more than he specifies will be retrieved.
   c. interactive
6. dictionaries
   a. exclusion - contains list of terms which will not be put into an index.
   b. synonym
7. index reference - the ability to inquire as to what values are in a particular index
8. save and re-use - the ability to name a search request or the results of a search and have the system store it. The request or results could be used later upon entry of the name assigned.
9. standing request - the ability to enter a retrieval request and have all new material added to a file compared with it. Any records meeting its criteria would be communicated to the user.
10. on-line recovery of search process - to insure that a user will not lose the results of an interactive search derived over a set of several interactions because of a temporary system failure.

The output facilities to be implemented are:

1. on-line
2. batch print
3. batch tape
4. formats
   a. system standard - formats specified by the system and available for anyone's use.
   b. file standard - formats specified by a file manager and available to any user of that file.
   c. user-defined - the ability for a user to specify a format while at the terminal.

5. sorting (for batch output only) - the ability to list retrieved records on a printer, ordered on the values of one or more data elements.
6. catalog cards - a printing, directly onto cards, of information contained in selected elements of a bibliographic file. This would be a batch operation.

The following training facilities will be provided:

1. reference manuals
2. reference cards
3. on-line aids - capability for the user to ask for help from the system through his terminal.

9.2 File Management

The following major facilities will be implemented:

1. definition of file characteristics
2. modification of file characteristics
3. buildup of file from initial data
4. updating
5. special listings - these will generally be unique to a file as specified by the file manager
6. statistical feedback
7. training

The file definition facility will allow the file manager to specify:

1. amount of required storage - ability to specify to the system the initial size of the file and its rate of growth.
2. data elements
   a. element name
   b. multiplicity
   c. element size
   d. data type - e.g., dates, personal names, numbers.
   e. choice of input editing
   f. hierarchical relations - for instance, one data element might contain a list of project names. Associated with each project is a data element which has a list of employees assigned to it. Associated with each employee is a data element which contains a list of tasks for him.
   g. automatic functions (to be executed upon occurrence of transaction for the element)

3. indexing
   a. which elements will be indexed
   b. addition, deletion of indexes
   c. editing of values to be indexed
4. dictionaries
   a. codes
   b. exclusion - the user should be able to override and force a term into the index for some records.
   c. inclusion - a list of words which will be put into an index. All other words will be omitted from the index.
   d. synonym dictionaries
5. display formats - ability to specify standard formats for the file. Each format would have a name which a user would enter in an output request. The format would specify the elements to be displayed and their order.
6. error severity level - the ability for the manager to specify the action to be taken upon the occurrence of various errors. The choice of actions includes: nullifying the user's request, presenting an error message and attempting to correct the error.

The following file maintenance facilities will be provided:

1. tape conversions
2. on-line entry of input and update requests
3. batch execution of updates
4. on-line execution of updates
5. update requests
   a. addition, deletion of records
   b. addition, deletion, alteration of elements or parts of elements.
   c. copy - from record to record and from file to file.
6. index of the record identification data element
7. applications - specific batch facilities will be provided on demand when feasible. These will normally be paid for by the user who requests them.
8. File merging and elimination of duplicates.

The training facilities which will be provided are:

1. reference manuals
2. reference cards
3. on-line help
4. consultation service

Miscellaneous Features

1. file specific message of the day
2. collection facility for user documents submitted on-line

APPENDIX K: Tutorial: Information Storage and Retrieval by James Marsheck

APPENDIX G

TUTORIAL: INFORMATION STORAGE AND RETRIEVAL

This appendix is intended to serve as an introduction to the concepts involved in the view of Information Storage and Retrieval held by the staff of the SPIRES/BALLOTS project. It is not a survey and does not attempt to cover all relevant problems or all of the techniques that have been developed in this area of computer technology.

A. TERMINOLOGY

In order to clarify the following introduction to the field of Information Storage and Retrieval, several key terms are defined. These terms are: files, retrieval, sequential files, direct access files, search and output. Other important terms are defined as they are introduced in the text.

A FILE is any body of information which exists on some storage medium and is structured so that segments of the information can be located and extracted in a systematic way. An example is a card catalog in a library. The storage medium is the cabinets containing cards and the systematic organization is an alphabetic ordering by author, title and subject. Another file, similar in structure though different in content, is the set of employee records stored in manila folders in a personnel office. A somewhat different kind of file is the multiple listing maintained by real estate sales firms. This file might be organized by price range, number of rooms or architectural style.

Once a file is established, the process of locating and extracting information is called RETRIEVAL. This process consists of several actions. The first is to formulate a QUERY, e.g., find the names of all books in the library pertaining to Serbian History. The second action is to look for relevant information. In this example, the inquirer scans the cards for the phrases 'Serbia-History' and 'History, Serbian'. The final action is to remove or copy the segments of information which satisfy the query conditions. In this example, removing the catalog cards, even momentarily, is not acceptable; therefore, the retriever would copy the information onto a loan request, charge slip or his own 3x5 cards.

Files are usually classified as SEQUENTIAL or DIRECT ACCESS although some might be considered a combination of the two. A SEQUENTIAL FILE is ordered in a single manner.

in order to locate any particular item of information, it is necessary to pass over to all preceding items.

In a DIRECT ACCESS FILE, any item may be retrieved without passing over a number of other items. To illustrate the difference, consider two files consisting of film representing a pictorial record of a vacation to Oregon. One of these files is a reel of 16 mm film and is a sequential file. To show Crater Lake, all of the scenes recorded prior to that must be passed over first. The second file is a set of 35 mm slides and represents a direct access file. To show the scenes of Crater Lake, only that specific set of slides need be projected. To locate the required set quickly, a list of scenes is maintained in some detail indicating which box or tray each set is stored in. This list is an index to the file. The concept of an index will be discussed later since it is central to the feasibility and utility of information storage and retrieval.

The process of locating the information described by a user in his query is called SEARCHING. The query is sometimes called a SEARCH REQUEST. The process of presenting the segments located by the search is called OUTPUT. Also, the resulting copy of the information is called the OUTPUT for the request. Both of these functions are discussed in later sections in more detail. Consider a search request applied to a personnel file to locate the records of all employees under 30 years of age earning in excess of ten thousand dollars. The computer, assuming a sequential file, examines the record of every employee in the file and checks the age and salary. This operation constitutes the search. For each record meeting the conditions specified in the query, the items of information in that record which were specified in the OUTPUT FORMAT (for instance, name, position and department) are printed. This is the output process for the example.

B. FILES

Files are stored on various media. Some of these are cards, sheets of paper, film and metal plates and are collected on shelves, in cabinets, in notebooks, on racks or in bound volumes. These files may contain many different kinds of information, as:

1. purely numeric items in a volume of statistical tables,
2. blueprints in an architect's file,
3. the textual content of an encyclopedia,
4. the mixed format of a personnel file.

The latter contains items which are numeric (age, salary), textual (references), coded (skill categories) and special forms (date of employment, inverted name).

Although most files not stored on computer equipment are sequential in nature, they usually have some of the characteristics of a direct access file. For example, an encyclopedia is organized by subject matter in alphabetical order. However, since each volume has the range of subjects printed on the spine, a person who is seeking information may narrow his search immediately to a specific volume. He then will find the correct page by making successive approximations and will have completed the entire search in a matter of seconds. The limitation of this technique is that the user of the encyclopedia must be familiar with the subject classification and often he does not retrieve all the relevant material. For instance, if he is looking for biographical material on Abraham Lincoln, he may not find the additional information contained under the subjects of Ulysses Grant or Appomattox.

Similarly, if a personnel file is ordered alphabetically on last name, it may be accessed quite efficiently when retrieving the records of individual employees whose name are known to the searcher. However, for any other type of retrieval, additional capability is required. This could be achieved by having multiple copies of the file, each of them ordered on some attribute of the employee, e.g., social security number, job classification, review date. Obviously, this would be too expensive and would lead to an unacceptably large number of errors. A more manageable alternative is to maintain a list for each category of information which INDEXES the file. For instance, a list could be maintained of all job classifications. Under each entry in this list would be a list of names of employees having that classification. If someone wished to send a memorandum to all executive secretaries, he could consult the list and obtain their names. From the file itself, he could get the company address for each.

The technique just described transforms an essentially sequential file into a form of direct access file. However, it is still somewhat cumbersome and prone to errors since, for each change in the file, one or more of the indexes may have to be changed. Another difficulty arises from the fact that the file exists in only one location while people in many locations may need to access it. Also, if one user of the file has removed a record, other users must wait until the record is returned. Many of the problems inherent in manual files can be resolved by placing them in the environment of a computerized information storage and retrieval system.

A sequential file to be accessed through a computer is normally stored on MAGNETIC TAPE. These tapes, and the mechanisms which write information on them (and read from them) are similar to home recorders, though larger, more complex and more expensive. A file on tape is purely sequential. It is restricted to a single ordering, and to access any one record, all previous records on the tape must be passed over. Another limitation of tape files arises from the fact that the tapes are normally stored OFF-LINE, i.e., on racks away from the computer. The information may be retrieved only when the tape is mounted on the read/write mechanism. Primarily because the tapes are stored off-line, this type of file is relatively inexpensive. It is a satisfactory mode of storage for files when the normal requirement is for large amounts of information on an infrequent basis rather than small amounts frequently and rapidly.

Computerized direct access files are normally stored on MAGNETIC DISKS. These disks are similar to phonograph records except that the recording is done magnetically rather than by physically cutting into the disk. The storage mechanism for direct access files is similar to the arm on an automatic changer. The disk access mechanism has the read/write cartridge on an arm which moves across the disk allowing rapid access to any track. Thus the information stored on a track of the disk may be accessed without reading over the information on other tracks. For instance, if each track held one employee record, then any employee record could be retrieved immediately if the numeric ADDRESS of the track for that employee were known. Having a sound method for determination of track addresses is one basis of a successful information storage and retrieval system of this type.

For the personnel file referred to above, retrieval requests will normally be stated in terms of employee attributes such as name, job classification, review date and skill categories. Other attributes such as home address and name of spouse are in the record of the employee but are not normally used in the formulation of queries. The attributes of the employees are called the DATA ELEMENTS of the file. The data elements which can be used in retrieval requests are called the ACCESS POINTS for the file. In a file of bibliographic references, the data elements would be items like author, title, publisher, number of pages and date of publication. The access points might be author, title and date of publication.

A means of creating access points for files is to construct an INDEX for each data element which is used for searching. The set of indexes is also stored on disks, in an order which allows efficient searching. An example is the AUTHOR INDEX for a bibliographic file. Assume that, on the average, the names of 50 authors can be stored on a single track of a disk and that the file contains the names of 2000 authors. The names are stored, in alphabetical order, over 40 tracks. In addition, a master track contains the first name on each track of the index. Each author's name has one or more addresses stored with it which indicate the location of each bibliographic reference associated with that author. If a user specifies the name Harrison H. Smedley in his search request, the following steps are taken by the computer. The master track for the author index is retrieved from a disk. The list of names in it is searched for a pair of consecutive names between which Smedley falls alphabetically. The address associated with the name which comes before Smedley is used to retrieve another track from the disk. If that track does not contain the name Smedley, the user is informed that the file has no references for Smedley. If, on the other hand, an entry for Smedley is found in that track of the index, the addresses contained in the entry allow the computer to retrieve all of the bibliographic references in the file for works authored by Smedley.

The organization of indexes in an information system is actually more complex than this but the general principle is the same. Records, whether bibliographic references, employee records or parts descriptions, have many data elements in varied formats. Because of this, ordering the file (i.e., the group of records) to facilitate retrieval is extremely expensive, if not impossible, even on the most powerful and sophisticated equipment. However, since each index contains only one kind of information it may be ordered relatively easily and in this way tailored to fit the type of data stored for that particular data element. For instance, dates may be indexed in chronological order or in reverse chronological order. Indexing does have economic limits. If many data elements are indexed, the total storage required for indexes may double or triple the amount required for the file itself. This is because of the relatively complex structure of the indexes. Disk storage is also more expensive than tape storage because the mechanism is much more complicated and costly to manufacture.

C. RETRIEVAL

Two examples of manual information retrieval are given as a contrast to computerized information retrieval. In the first example, it is desired to obtain from a personnel file a list of all employees who speak French, have a degree in electrical engineering, have at least two years of professional experience and are not married. The usual practice would be to submit a request for this information to a personnel clerk. This clerk would pull each employee record out of the filing cabinet, one at a time, and examine it to determine if that employee met the conditions of the request. For a large file, this would consume a large amount of the clerk's time in a purely routine task. If the file system is well designed, there might be a list of engineering employees which could be used to reduce the effort. If the personnel department is busy, the requester might have to wait several days to get his information. In addition, one or more employees who meet his requirements might be missed due to human error.

A second example illustrates a retrieval process which is often more wasteful and prone to inaccuracy than the one in the first example. Assume that a medical research scientist wishes to propose the initiation of a new project investigating the effects on human metabolism of the prolonged use of artificial sweeteners. He does not wish to duplicate work which is complete or in progress so he requires information on recent projects in this area. There are several resources he can use in attempting to get this information.

First he can scan all of the applicable journals published during the years he is interested in. Secondly, he may consult his associates to determine if they know of any relevant research. Thirdly, he can contact the leading research organizations to inquire about their current and recent projects. Also, there may be a review published which covers a significant portion of the field. Several major difficulties are inherent in this procedure. It could take several weeks to complete the survey. Several hours effort of highly skilled people is involved. The probability is high that some significant research will be overlooked. A significant amount of the research budget might be consumed in carrying out a function which does not contribute directly to research results.

These difficulties can be alleviated by the use of computerized information storage and retrieval systems. However, it is not necessary, and perhaps not desirable, to have all retrieval functions performed by computer. The user of the system can often benefit, both in terms of the effectiveness and of the economy of retrieval, by having some operations performed manually or by non-computer equipment in conjunction with the computer system. Consider, for example, a bibliographic file, including abstract material or even full text on microfilm. Indexes for the file can be maintained on a computer. The user can then carry out his search through the computer, receiving as output a list of numbers referencing the microfilm which is stored either in cabinets or on special equipment designed for that medium. He might then use a microfilm reader to scan the abstracts and select a final subset of documents. Finally, he or a library assistant would make hard copies of the documents.

The way in which a computer is used to retrieve information from a file depends on several considerations. The first is the frequency with which people request information. Are there several inquiries per day or several per minute? Another consideration concerns the amount of material to be retrieved. Is it normally a yes or no answer (do we have any widgets in stock?), a single name or quantity, a short list of employees and their review dates or a large amount of information such as an address list. A third point is response time: are answers usually required in minutes, hours or days?

The complexity of an inquiry is an involved question and affects, for instance, the way the query is expressed. A SIMPLE REQUEST might be expressed in a single employee name or parts number. A more complex query might be stated in a form which indicates several conditions are to be satisfied before an entry in the file is retrieved. For example, the request "FIND ALL EMPLOYEES WITH SALARY GREATER THAN 10,000 AND AGE LESS THAN 30 AND WITH CLASSIFICATION PROGRAMMER" will return the records of all employees who are programmers under the age of 30 earning more the 10,000 dollars and no other records. This format for a request is called a logical expression.

Another consideration is the complexity of the output. A very simple output consists of every data element in a record, listed in the order it is stored in the file, with one data element per line. A slight complication is introduced if the user specifies that some subset of the elements be listed in a particular order. A sophisticated output facility allows the user to specify page format, i.e., margin size, columnization, double spacing, etc. Some users of the system may require that output be sorted on one or more data elements. For instance, a retrieval request might be for all employees who have an imminent review date with the output listed in order of department number. Often, it is desirable to obtain statistical information on a file which introduces another kind of complexity to the output. For example, what is the average relocation expense claimed by employees hired during the past year or what is the maximum and average number of citations retrieved from the physical science section of a bibliographic file during the last two months.

There are two quite different ways in which a user can communicate with a computer in retrieving information from a file. The first, called BATCH processing, is used when:

1. single requests are for large amounts of information,

2. a response time measured in hours or days is acceptable,
3. output requirements are very complex.

The normal manner of operation for BATCH RETRIEVAL is as follows:

1. a query is formulated and punched on cards,
2. the cards are submitted to a computer operator,
3. he schedules the query and places the cards in a batch with other request cards,
4. the search is executed at the scheduled time (often overnight) and output listed on a high-speed printer,
5. the listing is delivered to the requester.

A purely batch retrieval system is relatively easy and inexpensive to implement but has some definite limitations.

However, an ON-LINE system should be used if the users of the system require answers in minutes or need help from the system in formulating their request, i.e., the first try does not retrieve the material desired and one or more re-formulations must be attempted. In an on-line system several users are communicating with the computer simultaneously. This is accomplished by having many terminals connected to the computer in much the same way that many telephones are connected to a switchboard. In this mode of operation, a retriever enters his request through his terminal and receives a response almost instantaneously. If the request requires a long search, the initial response may be only an indication that the request has been accepted and the computer is in the process of executing it. It may take as long as several minutes to return an answer to some requests. The time that elapses between entering a request and receiving a reply is usually called response time. The elapsed time between receiving a response and entering the next request is normally called think time. People read, reason, and type slowly, in comparison to machine operation time. Think time tends to be fairly long relative to execution time. Thus, the on-line system is able to execute requests for several other users while a single user is digesting the answer to his request.

Basically, there are two types of computer terminals. One type is simply a modified electric typewriter with a wide carriage, a few special function keys and a connection (often a regular telephone line) to the computer. The other type is a screen, similar to the visual part of a television set with a small keyboard added. This kind of terminal is usually called a CRT (short for cathode ray tube) and the output from the computer shown on the screen is called a

DISPLAY. The advantages of a typewriter terminal are: it is relatively inexpensive and it provides hard copy. The disadvantages are: it is relatively slow, it is noisy (especially if several are clustered in one location) and it requires more effort from the user. The advantages of a CRT are: it is virtually noiseless, it is relatively fast (some models can display hundreds of characters in the blink of an eye), and it can be used in ways that make man-machine communication very efficient and effective. The disadvantages are: it provides no hard copy and is expensive. It is possible to combine typewriter and CRT into one terminal and gain a great deal of flexibility but the cost is greater than either device alone.

In many cases, it is not desirable to have a purely batch or a purely on-line information system. Fortunately, there are several ways to combine the two concepts into a single system. The simplest solution is to have an on-line system going during the day and a batch system during the night shift. A more sophisticated solution and one which allows more efficient use of the computer and gives more flexible service to the user community is a system which handles both on-line and batch requests simultaneously. The on-line part of the system has priority and all requests from terminals are satisfied as they are entered. However, the computer frequently runs out of requests to execute and waits for a message to be entered from some terminal. During this wait time, the batch part of the system is given control of the computer and processes part of the batch workload. When a terminal request is entered, control reverts to the on-line part of the system. The batch system is operating in what is called BACKGROUND processing.

As indicated above, both a query and the resulting output can range from very simple to very complex. In order to clarify a discussion of various kinds of retrieval, a brief outline of a session at a terminal follows. The first step that the user takes is to sign on, or "Log On", to the system. This consists of turning on the device and waiting for a signal that the computer is ready for communication. In some cases it is necessary to dial the computer's 'phone number'. The user then keys in a few pieces of general information like his name and account number. The next step is usually the selection of one of the available files. The system then responds with a PROMPT (questions from the computer are called prompts) indicating that it is ready for the user to enter a query.

The user then formulates his query, and types it in. When he hits some particular key (on a typewriter, this is probably the carriage return) the computer examines the message. If it detects an error or does not 'understand' the request, an error message is returned along with a prompt for him to re-enter the query. If the request is correctly formulated, it is placed in a queue (waiting line) and serviced in turn. The queries (and other requests such as output format) are expressed in a language which contains a very limited set of English words and uses a very simple grammatical structure. Since the prompts are considered part of this language and the communication is two way, this language is a CONVERSATIONAL or INTERACTIVE language. Requests directed to a batch system, on the other hand, do not normally have this property.

When the system completes the requested search, it types or displays some response. In the case of certain simple kinds of queries, this message is the requested information. In other cases, the system informs the user of the number of items which meet his CRITERIA (the conditions stated in his query) and waits for him to enter his next request. The user then decides if he wishes to see the information in the retrieved records or if he wishes to refine the criteria and enter a request that will be combined with the previous one to enlarge or reduce the set of retrieved records. An additional step may then be taken; some users will ask for a listing on a high speed printer if he has many pages and wishes to keep a permanent record of his retrieval. The printer is able to list several hundred lines per minute with each line having as many as 133 characters. Also, the printer operates in the background mode and is much less expensive.

The relative simplicity or complexity of retrieval requests, in terms of search and output, determines:

1. the choice of terminal,
2. the way in which files are indexed,
3. the facilities provided for search and output in both the on-line and the batch parts of the system.

For the simplest variety of request, the query contains only the identification of one data element and a single value for it and the output is simply the value of another data element for any record meeting the single criterion. An example of such a request is: RETRIEVE EMPLOYEE JOHN Q. SMITH; OUTPUT SALARY. The system would search the index for employee name, locate the record for John Q. Smith and type or display his salary. For this type of request, there is little difference between a typewriter terminal and a CRT except the cost of the equipment. The complexity increases very little if several items are combined into a LOGICAL EXPRESSION in the search request and more than one item is requested in the output, as: RETRIEVE JOHN Q. SMITH AND HARRY P. ANDERSON; OUTPUT SALARY, POSITION, AGE. There are two distinguishing characteristics of this form of retrieval. The user is able to supply information to retrieve an explicit subset of records from which he requires information. The information he wishes to see is contained in a small number of records in an easily extracted form and he wishes it to be presented essentially as it exists. The principle requirement in this kind of retrieval is that all the data elements which can be specified in a search request must be indexed.

For a contrasting example, consider the query,

FIND ALL TITLES SPIRIT, GHOSTS OR APPARITION,

applied to a file of bibliographic references. The system searches the index for the title data element, locates all references containing any of the three given words in the title and responds with a message indicating how many references have been found, say 46. He then enters the request: OUTPUT TITLE. Suppose the first three titles to be presented were:

The Problem of Ghosts on Television Screens
The Spirit of Christmas
Apparition and Mysticism in Religion.

To reduce the amount of unwanted references in the set he has retrieved, the user enters a modification to his search request: BUT NOT TITLE TELEVISION OR CHRISTMAS OR RELIGION. This might reduce the set to include only relevant material or he might have to make further modifications to the search request. In addition to the problem of retrieving unwanted information, there is also a possibility of not finding some relevant material. There are two things which can be done to alleviate these problems.

Much of the problem of unwanted or lost information is caused by the variety and ambiguity of words in the English language. A contributing factor is that the titles of most books and documents do not reflect completely and accurately the contents. Therefore, searching on the basis of title alone is not an adequate retrieval technique. If a bibliographic file is constructed with a data element that contains phrases descriptive of the subject matter in a document, this data element, when indexed, will usually be useful in retrieval. This type of index is usually called a TOPIC, SUBJECT or KEYWORD index. In addition, an information retrieval system should provide a thesaurus capability. By using a thesaurus a user is able to determine the phrases which are used to describe a topic. He also receives help in formulating his request in a way which helps ensure the retrieval of all relevant material. For instance, if he consults the thesaurus under the word ghost, he might receive the response: SEE ALSO POLTERGEIST.

A third type of retrieval usually has a fairly simple and explicit request in terms of the search but a complex or lengthy requirement for output. For example, in accessing a parts inventory file, to find all parts which are out of stock: RETRIEVE ALL PARTS,

> STOCK = 0; LIST NAME, PART NUMBER, ORDER DATE, AVERAGE MONTHLY SALES, PRICE; ORDER ALPHABET (NAME).

This request might be entered either through a terminal or, on punched cards, into the batch system. Because of the requirement to sort the output, it would be executed by the batch system. In this example, if there was an index on the data element STOCK, an entry in that index would contain a list of the locations in the file of the records of all parts which were out of stock. Each of these records would be retrieved, the data elements specified for output extracted and an intermediate file created, probably on disk.

This intermediate file would be used as input to a sort program which would produce the output on a high speed printer, ordered alphabetically by part name. If no index existed for the data element STOCK, the batch retrieval would have to read every record in the file and check for a zero value for STOCK.

When a file is set up, a choice is made of the data elements which are to be indexed. Since an index requires a significant amount of storage and adds processing time to the file maintenance, an evaluation is made of the frequency with which that data element might be used as an access point. This helps determine if the cost of the index is justified by expected savings in the processing of queries.

A second example of a retrieval request with output requirements that demand extra processing is the query to a personnel file:

> FIND ALL EMPLOYEES, POSITION SECRETARY; OUTPUT AVERAGE AGE, SALARY RANGE, AVERAGE SALARY.

For this request, the system locates the records for all secretaries, computes the average age and salary and lists them along with the lowest and highest secretarial salary. This request could be processed by either the on-line or batch system since the computation is a fairly simple operation.

D. FILE MANAGEMENT

An information storage and retrieval system can support a number of files. For each of these files, there must be someone who is responsible for its management. The person who assumes this responsibility is sometimes called a FILE MANAGER. His tasks include:

1. estimating the size of the file,
2. deciding whether it is to be a direct-access on-line file or a sequential file,
3. specifying the data elements and the indexing requirements,
4. determining who is authorized to access the information contained in it,
5. providing the data for the initial file buildup,
6. supervision of the people who maintain the file.

FILE MAINTENANCE is the process of:

1. adding, deleting and modifying records in the file,
2. editing data to ensure the reliability of the information,
3. initiating the use of backup facilities,
4. executing recovery procedures when damage occurs to the file.

A BACKUP facility provides the ability to make copies of the file on magnetic tape and to maintain a log of recent changes or additions to the file. Together, these may be used to restore a file when some information has been lost or damaged due to computer, program or human malfunction.

The first task of the file manager is FILE DEFINITION, which is the process of specifying the FILE CHARACTERISTICS. Great care should be taken in defining these characteristics since many of the choices made at this time can seriously limit the information which can be put into the file. These choices may restrict and hamper file maintenance tasks. The file manager should take advantage of any consulting services which are offered by the SYSTEM MANAGER, who is responsible for the design, development and maintenance of the information system itself. He may also be in charge of the operation of the computer and related equipment. In fact, in some organizations, his title might be operations manager.

The items which must be specified in the file definition are: the data elements, the properties of the data elements, indexing requirements, thesaurus facilities, display and report formats, editing requirements, partitioning criteria, backup needs and security requirements. Each data element is given a name which is used in the remainder of the definition specifications, in retrieval requests and in output requests. Many systems also allow abbreviations and synonyms for data element names. Other properties to be specified for data elements are DATA TYPE, maximum length and multiplicity. Data type describes the kind of information contained in an element, e.g., numbers, dates, names of people, codes or text. The MAXIMUM LENGTH is the largest number of characters which any value of an element may have and it is used in checking the input data for errors. MULTIPLICITY is simply an indication of whether or not the data element may have more than one value for any given record in the file. Examples of singular data elements are employee name and publisher's address; examples of multiple data elements are languages spoken by an employee and authors of a book.

After considering the various needs of the people who will be retrieving information from the file, the manager must specify the indexing requirements for the file. The first consideration is: which data elements are to be used in expressing search requests? Each of these elements must then be indexed. In addition to indicating the elements to be indexed, he must select which editing facility will be applied to the values in that index. Consider, for example, the title index of a bibliographic file. There are several editing functions which the manager may wish to have performed on titles as they are indexed. First he may wish to delete special characters, such as commas, quotes, periods and colons. Secondly, he may specify a DICTIONARY of words like "IT", "THE", and "A" which should not be indexed. This dictionary is often called an exclusion list; if prepared carefully, it can save considerable storage and processing costs.

For bibliographic files, the manager must specify the contents of a THESAURUS for that file since the words and their relationships are dependent on the subject matter of the file. The thesaurus entry for a word (or a phrase) may have a list of synonyms for that word which helps the user in retrieving further relevant material. It may also show hierarchial relations with other words, i.e., words which are more specific or more general in nature but concerned with the same topic.

While the system will provide some standard formats for display of information on terminals and for listings to be produced on high speed printers, some file managers may wish to specify special formats tailored to the needs associated with their own files. The specification of editing requirements, partitioning criteria, backup needs and security requirements will be described in the appropriate paragraphs below.

The second major task of the file manager is to acquire the data which constitute the information in the file. This data may exist in any of several forms, e.g., file cards, printed material, punched cards or magnetic tape. It may, as in the first two cases above, have to be converted to a form which can be read by the computer. If the data is on cards or magnetic tape, a computer program may have to be written which alters the format so that the input programs of the information system can handle it. Finally, the file manager will have to initiate, with the assistance of the system or operations manager, the process of file building. This normally consists of punching a few system control cards and delivering the input data to a dispatch clerk or a computer operator.

Maintenance of the file includes the functions of adding new information (bibliographic references for recently acquired books), deleting or purging obsolescent material (the records of terminated employees) and the modification of information, (correction of spelling, salary raises, change of address, updating of inventory). For reliability of the file, it is necessary to edit the information as it is input and to provide for backup and recovery. Some editing may be done by the system but much of it can often be done only by manual means. For instance, the computer can be programmed to recognize that JAN 41, 936 is not a legal date but not that the "e" was left off of the name Johnstone. Unfortunately, there are occasions when a computer malfunction or a programming error will cause some information in one or more files to be altered or destroyed. In order to prevent this from becoming a disaster, an information system must provide facilities for backup and recovery. The most common technique used for this purpose consists of periodically copying the file onto a magnetic tape and storing it out of harm's way. In addition, a TRANSACTION FILE is maintained (probably on tape also) of all changes to the file (additions, deletions, etc.) since the last backup was executed. Thus, when damage occurs to an on-line file, recovery is achieved by restoring it from the last backup tape and re-executing the recent changes.

One more very important responsibility of the file manager is prescribing the availability of the file. It may not be economically feasible to have the file on-line all the time the system is operational. So, he may decide to make it available for retrieval only during certain scheduled hours. At other times the disk(s) containing the file can be stored away from the computer. This will free part of the computer equipment for use with other files. Since the access mechanism itself is much more expensive than the disk, a significant savings can be achieved this way. A second availability factor concerns who is able to retrieve from the file. Some files may be public in that any one who has a terminal and an authorized account number may access them. Others may be private with only the file manager and his associates permitted to retrieve information from them. To support this restricted accessibility and to prevent unauthorized persons from altering information in a file, the system must provide a security facility. This usually involves the specification of PASSWORDS by the manager. A user must then know a password to access a private file or to alter the contents of any file.

APPENDIX L

SPIRES II SHARED FACILITIES

(Excerpt from "System Scope for Library Automation and Generalized Information Storage and Retrieval at Stanford University")

10.0 SUMMARY OF CURRENT SHARED FACILITIES

10.1 General Concepts

DEFINITION

Shared facilities consist of software and hardware designed to provide concurrent service to functionally related applications.

ECONOMIC CONSIDERATIONS

A gross estimate reveals that in terms of implementation effort, SPIRES/BALLOTS II may be broken down approximately as follows:

... BALLOTS - 1/3
... SPIRES - 1/3
... Shared facilities - 1/3

If each application user pays for his own development plus half for the shared facilities, that user effectively gets the use of sixty-seven percent of the system for half the total investment. Alternatively, if two users invest similar amounts in separate development efforts, each is given substantially less for his money. Another operative factor is hardware economy of scale. If two users pool their resources to acquire shared hardware, the resulting individual capability will be greater than it would with separate installations. This simple analysis argues for continuing combined SPIRES/BALLOTS development.

10.2 Present Shared Facilities

COMPUTER OPERATIONS ENVIRONMENT

SPIRES/BALLOTS I software executes on an IBM 360 Model 67 located in the Campus Facility of the Stanford Computation Center. This computer has one million characters of main storage, and processes data input and output through ultra-high-speed and high-speed direct-access devices as well as magnetic tapes, card equipment, and line printers.

Installation software and procedures are directed toward a rapid throughput computation-oriented market. Although the data processing facilities provided are of excellent quality, high priority is placed on keeping the computation facilities operative. If a file failure occurs, correction must wait until a scheduled software maintenance interval. This could result in an unacceptable inconvenience to the non-standard user who has very large, continually updated files.

There are two pieces of computer memory available for program execution. The first is approximately 100,000 characters long, and will accept no job whose duration exceeds two minutes. The second is approximately 300,000 characters long, and will accept jobs of any duration. SPIRES/BALLOTS I uses the latter. A great disadvantage is that while someone else is executing in this portion of memory, SPIRES/BALLOTS cannot and vice versa. This precludes extended, exclusive use of the computer resources by SPIRES/BALLOTS I.

The policy in this operations environment is to discourage long-duration jobs by charging them more per execution minute as the job progresses in time on the computer. A further discrimination is made between day and night jobs; it is cheaper to run at night. It is clear that these policies are not constructed to benefit a system such as SPIRES/BALLOTS I. A further problem is a lack of guaranteed access to the system from a terminal; there are over 200 terminals connected to the system and only 60 can be in use simultaneously.

The model 67 is currently approaching its capacity, at least during peak periods. These periods occur near mid-term and final examination time or roughly eight times per year. During such intervals the execution backlog grows long, and it is difficult to gain access to the system through a terminal.

ON-LINE EXECUTIVE PROGRAM

The SPIRES/BALLOTS I Supervisor is an on-line executive program designed and developed by project personnel to service several on-line users simultaneously. The purpose of an on-line executive program is to regulate the competition for service and resources among several terminal users. The program attempts to insure that each user gets a reasonable share of available execution time. Experience with the SPIRES/BALLOTS I supervisor has demonstrated the feasibility of the approach taken; response time averages three seconds for simple search requests.

TERMINAL HANDLER

The terminal handler performs the actual input/output operations between remote terminal locations and the main computer. Its role is that of a middleman standing between the terminal lines and the on-line executive program. This function is currently discharged by MILTEN, a program provided by the Campus Facility installation. Part of the program resides in the main storage of the Model 67, and the rest in a smaller computer (PDP-9) to which the terminal lines are attached.

ON-LINE DATA COLLECTOR/TEXT EDITOR

The purpose of this program is to allow the on-line collection of input data for later use by batch computer runs. It further allows correction and modification of such data at the character level. This facility has been found extremely useful in gathering data to be used in file building; most users have chosen it in lieu of punched cards and found it easier and cheaper than less flexible alternatives.

The need for a Data Collector/Text Editor is currently satisfied by WYLBUR, which is part of the Campus Facility installation software. It has been found to be excellent in all respects save one: it requires the user to backup his files, rather than provide such service automatically.

FILE SUPPORT

The basis for any information storage and retrieval system is the collection of files it handles. These files may or may not have any connection among themselves. For example, the entire collection may contain files related to personnel records, medical data, or bibliographic data concerning published documents. There is no restriction on the information that can be stored and no two distinct groups of files need have a relationship.

Files within the collection that are connected or related to one another in some predetermined way are defined to be a set of related files. The system supports two types of related files: principal and statistical.

Principal files serve as the basis of operation for the user within the system. In these he accumulates his primary data: texts, abstracts or other data elements, their associated access indexes, and file characteristics.

Statistical files contain information on the contents and usage of corresponding principal files.

RECOVERY/RELIABILITY

The Campus Facility System fails at least once every 36 hours, and sometimes more often. The incidence of failure may seem high, but realistically speaking, the system has excellent reliability for such a complex collection of facilities. Such failures, however, can cause an unacceptable loss of a large continually updated file.

Recovery of files whose integrity has been lost in such situations is accomplished by periodically copying the file to magnetic tape (called dumping) and recopying back to disk (called restoring) following the failure. It has proved economical to dump a file after each one-hour aggregate of file building time.

AVAILABILITY

The current SPIRES/BALLOTS files are available during the day and most of the night. The on-line executive program, however, is not. At the present time, there is no regularly scheduled SPIRES/BALLOTS service block, and users must bring SPIRES/BALLOTS into execution themselves. As discussed above, they pay premium prices as a result.

11.0 LONG-RANGE SCOPE, SHARED FACILITIES

BALLOTS and SPIRES will share common software/hardware facilities. It is difficult to predict the nature of application areas to be added in the future. In theory, any new application requiring on-line storage and manipulation of data can be accomodated. A necessity therefore exists to implement all shared facilities in a generalized, modular fashion to facilitate additions at the application level.

With the exception of added utility programs, there will be little expansion of shared facilities beyond the SPIRES/ BALLOTS II effort. Applications added later will be designed to interface with SPIRES/BALLOTS shared facilities, and will cause few perturbations at the shared facility level.

It follows that the long-range scope is identical to the scope for implementation in 1970-71.

12.0 FIRST IMPLEMENTATION SCOPE, SHARED FACILITIES

Below is a list of those facilities whose sharability is certain. As the detailed analysis and general design phases proceed, it may become apparent that other facilities may be generalized and shared (e.g., a batch update that works for both library and GISR users). Since no certainty now exists with regard to such facilities, they are treated separately in the two preceding sections.

COMPUTER OPERATIONS ENVIRONMENT

The operations environment for SPIRES/BALLOTS II will be a Data Facility. The hardware chosen will be only large enough to service present applications, with later augmentation as growth dictates. Procedural orientation within the facility will emphasize data handling rather than computation. High priority will be placed on the recovery of lost data as well as resumption of service to other users.

The Data Facility will handle long-duration and non-terminating jobs as well as short-duration utility jobs. There will be a greater guarantee of access to the machine during normal working hours, and machine resources will be provided once access is gained. Since the pressure of dominant, cyclic workloads will be absent, access contention will exist only within the data facility user group.

ON-LINE EXECUTIVE PROGRAM

All services provided by the SPIRES/BALLOTS I Supervisor will be provided in SPIRES/BALLOTS II. Design goals will include a maximum of flexibility and generality to facilitate the addition of new applications. Another desired feature is changeability of the user command language without resort to reprogramming. The language must be augmentable through the addition of new applications as well as changeable to whatever new experience dictates.

TERMINAL HANDLER

All services now provided by MILTEN running in the Model 67 and the PDP-9 will be provided by the new system. This could happen through the adaptation of MILTEN or some other pre-existing package to the new environment.

One additional condition to be met is the accessability of the data facility not only through new data facility terminals (CRT's, CRT's with hard copy, and 2741 typewriters) but also through the present campus communications network (2741's presently installed and hooked to the Campus Facility).

ON-LINE DATA COLLECTOR/TEXT EDITOR

All facilities now provided by WILBUR will exist as part of the new shared facilities. As with the terminal handler, this could happen through the adaptation of Campus Facility software, IBM software, or some presently unknown alternative. An additional feature will be the use of the text-editing capability in conjunction with on-line updating of data files.

FILE SUPPORT

The system will support, in addition to the principal and statistical files mentioned in 10.0, two other file categories: historical and holding.

Historical files are of two types. The first includes accumulations of transaction records that have updated principal files. Their role in file recovery is described below. The second type captures records deleted from principal files. This provides an alternative to the re-keyboarding of deleted records when their reuse becomes desirable. Both types of files will generally be retained as magnetic tape files.

Holding files are temporary files of data selected from principal files. These will fulfill the input requirements of scheduled batch processes or satisfy individual standing requests from users for selective reporting.

RECOVERY/RELIABILITY

Since files are the basis of the system, their reliability is extremely important. Information should not be irrecoverably lost or damaged in any way by user error, machine malfunction, or program problems. Should a file become damaged or destroyed, a set of methods must exist for immediately re-creating an image of the file as it was just prior to the malfunction, and quickly restoring service. The following discussion describes two techniques that will be used to achieve this.

1. SIMPLE COPY/RESTORE At specified intervals, a set of files is copied to magnetic tape. If the on-line version of those files suffers damage or is lost, the magnetic tapes can be recopied back on-line, thus restoring the files to their status as of the last copy to tape. In cases where few updates have occured in the intervening period, this method may be sufficient providing absolute file integrity is not required.

2. COPY/RESTART This method is similar to the simple copy/restore, with one enhancement: the history file, containing all adds, deletes, and changes to the file since the last copy, will be used to update the restored version to the condition of the file just prior to the malfunction. This is done when a file has undergone many changes since being copied to tape, and absolute file integrity is required.

AVAILABILITY/SECURITY

The availability of file sets has several aspects: service hours, public vs. private files, multiple users of files, and file security. All file sets and all information within those sets are not available to everyone at all times. Some files may be available for on-line retrieval at specified times during a day (if those files are on-line during that time) and perhaps available for batch maintenance at some other time. Other files may be concurrently available for retrieval and maintenance, implying on-line maintenance. There may be another category of files which are kept off-line and only placed on-line at the request of the user.

The availability of files to the user community also depends upon the status (public or private) which has been defined for those files. Public files can be accessed by anyone who desires to obtain information from them. Some large public files contain information received from a national bibliographic service via magnetic tape . A file may belong to a particular user who maintains the file and has complete responsibility for it. Such a file may be termed a personal file and still be available publicly, e.g. bibliographic data regarding a professor's private library. Private files can be accessed by a restricted number of users, possibly only the person responsible for that file. There are several variations on the public/private concept. Access to a file may be unrestricted; changing data within the file may be restricted to one or a few persons and still allow unrestricted query. Alternatively, access to a file may be partially restricted such that only a portion of a file or a certain set of data elements is available to general users.

There may be several users of the entire system at any one time. If a file is available to more than one user, there may be two or more users accessing information from the same file simultaneously. One user is not refused access to information in a file because information in that file is already being accessed by another (unless both users are attempting to update at the same time).

The ability to maintain files as public, private, or semiprivate is dependent upon a file security facility. Security must exist at these levels:

1. Files must be secured against access by anyone not having authorization.

2. Specified data elements within a file must be secured against access by anyone not having authorization.

3. Files must be secured against modification by anyone other than the file manager or persons given authorization by him.

Security at all levels could be effected through the use of group or individual passwords. A password is a string of characters which has been specified by the file manager as a key to gain access to his file. A searcher not responding to a request for the correct password would be denied his request for information retrieval. Other implementation possibilities include user definition of a security algorithm approoriate to a particular set of files.

ACCOUNTING

It will be necessary to design and implement accounting software to gather information for customer billing. This software must be sophisticated enough to distinguish between a user whose support requirements are small, and one who has complex requirements. Customer charges must accurately reflect machine resources actually utilized. With the exception of overhead rates, there will be no hidden subsidy of expensive facilities by customers not actually using them.

Such software is difficult to implement. This fact is reflected by a general lack of vendor accounting support until recently. In spite of this fact, it may be possible to adapt software developed elsewhere for this purpose, such as the System Management Facilities package distributed by IBM.

CHARACTER SETS AND SYMBOL REPRESENTATION

The capability will be provided to display or transliterate special symbols; for example:

... Mathematical symbols
... Symbols used in the physical sciences
... Greek letters
... Diacritical marks

Wherever direct display is not feasible, a notation such as 'A = *ALPHA*', could be used.

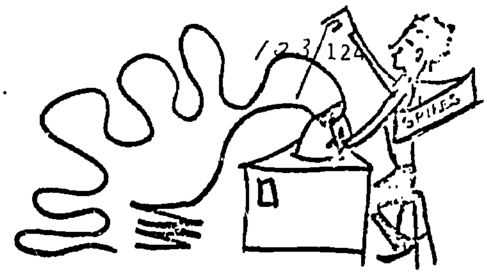
REPORT GENERATION

The capability (consistent with security) to select, format, and list data base elements will be provided on a batch basis.

APPENDIX M: Search Guides for PPF and IPF

S P I R E S

SAMPLE PREPRINT SEARCH *



Sign-on

NAME? *Physicist X*
ACCOUNT? *XXXX*
KEYWORD? *XXX*
TERMINAL? *XXX*

COMMAND? *SPIRES*
Welcome to SPIRES. For information use the SHOW NEWS command.

Specify Data Collection

SEARCH? *PREPRINT*

Author & Date Search

FIND? *AUTHOR gell-mann OR AUTHOR r. feynman*

author search for Gell-mann
author search for R. Feynman
20 document(s) accumulated

? *DATE AFTER 1/68*

date search after Feb-1-1968
2 document(s) accumulated

? *TYPE LENGTH 80*

. . .

OPTION? *RESTART*

Citation Search

FIND? *CITATION PHRVA,62,85*

citation search for PHRVA,62,85
5 documents accumulated

? *date from May 1968 thru 1 dec 1968*

. . .

0 document(s) accumulated

BACKUP? *no*

Title word Search

FIND? *title rho# and not title pi and not title pi-#*

. . .

20 documents accumulated

? *not ti pion#*

. . .

15 documents accumulated

? *author Anderson*

. . .

2 documents accumulated

? *backup*

search results reset to last 15 documents

? *type length 90*

.. .. ..

OPTION? *to spires - gripe, gripe; praise, praise. signed*
TO SPIRES? *Physicist X.*
TO SPIRES? *[carriage return]*
OPTION? *logoff*

NOTES

Each new search begins with a FIND?

The ? indicates continuation of the previous search.

The OR logic broadens the search.

The AND and the NOT logic, narrow the search

AND is understood after ? unless OR is stated.

Default line length for a listing is 120 char. For an 8-1/2" page width, specify TYPE LENGTH 80.

To interrupt a listing, push ATTN button.

To continue in the same search after an interrupt, respond OPTION? SEARCH

To start a new search respond: ? RESTART or OPTION? RESTART

BACKUP takes you one step back within your current search.

TO SPIRES lets you send a message to us.

LOGOFF ends the session.

*For a much more detailed description of the SPIRES search language, consult the SPIRES REFERENCE MANUAL. Copies are available in the SLAC Library, x2411.

SPIRES PREPRINT SEARCH 6/69

Quick Guide

Contents of PREPRINT "data collection"

1. Bibliographic information and journal citations for 5000 experimental and theoretical high-energy physics preprints received in the SLAC library since March 1968. (We're also including instrumentation preprints since 1/69.)

   The data base is updated (with "next week's" preprints) each Thursday evening (barring human or electronic disaster).

2. Bibliographic information for all SLAC reports, pubs, and trans. Citations are entered for pubs dated later than Apr. 1968.

TITLE TERMINOLOGY

The current character set lacks Greek letters, superscripts, subscripts, and other special characters. The following are a few of the commonly used substitutes:

pi-minus	pi-plus-minus
pi-zero	anti-K
pi-plus	anti-p
rho-minus	K-L-3
K-plus	He-4
LAMBDA-plus	anti-e-neutrino

etc.

(If you've been reading the "PPF" preprint list which has been produced from the SPIRES data since Jan 1969, you'll be familiar with most of these conventions.)

UPPER & LOWER CASE

are ignored in search language. Commands may be given in all lower, all upper or any combination.

CONTINUATION

@ at end of line continues statement to next line.

SEARCHES

	Abbrev.
AUTHOR	(A)
TITLE	(TI)
DATE	(D)
date after	
date before	
date from -- thru --	
CITATION	(C)

LOGIC

AND

AND NOT

OR

OPTION?

SEARCH
RESTART
TYPE (short form output)
LENGTH ## (for 8-1/2" width, use 80)
TYPE EXTENDED
SHOW NEWS
SHOW OPTIONS
TO SPIRES
TO OPERATOR
EXIT
MILTEN
LOGOFF

?

BACKUP
(ALL SEARCH COMMANDS)
(ALL OPTIONS)

BACKUP?

YES (resets to previously collected group of documents)

NO (starts new search)

SPECIAL FEATURES

\# used as truncation signal with author's names and title words. Must be preceded by a minimum of 3 characters.

AUTHOR SCHWAR#
TITLE PI-#
TITLE PHOTOPROD#

" " may be used to enclose reserved words for searching. For example, C is reserved as the abbreviation for citation. To search for title C use TITLE "C".

AUTHOR'S NAMES

may be written

FIND? author J. Smith
FIND? author Smith, J.
FIND? author J.A. Smith

(each example will find James A. Smith and John Smith, etc.)

DATE

may be written almost any way:

12 Jul 1968
7/12/68
7-12-68
7-68
July 12, 1968
etc.

CITATION

must be written

Journal, Volume, Page

FIND? C PHRVA, 140B,1686
where PHRVA is a standard five letter CODEN for the Phys. Rev.

Nuovo Cim.	NUCIA
Phys. Rev.	PHRVA
Phys.Rev.Let.	PRLTA
Phys. Let.	PHLTA
Nucl. Phys.	NUPHA

For other commonly used CODEN, see Ref. Manual or call SLAC Library, x 2411.

NOTES ON CITATION SEARCHING IN THE SPIRES "PREPRINT" DATA COLLECTION

A citation search in SPIRES enables you to find recent papers which cite an earlier journal article. For instance, you can locate all the recent preprints which cite your Phys. Rev. Letters article of January 1968.

For a SPIRES citation search, you will need a bona fide journal reference. (Sorry, no proceedings, books, or preprints.) The search statement is:

```
FIND?     citation PHRVA,168,1858
                   ↑      ↑   ↑
                   |      |   └First page
                   |      └Volume
                   └Five letter abbreviation (CODEN) for journal
```

The example above will locate all preprints which have cited:

> A. Pais and S.B. Treiman, "Pion Phase-Shift Information from K-L-4 Decays," Phys. Rev. 168, 1858 (1968).

A few commonly used CODEN abbreviations are given on the "Quick Guide" sheet. A more complete list is included in the SPIRES REFERENCE MANUAL.

The citation search frequently provides an excellent subject approach to the preprint collection. It is important, however, to choose two or more key articles which are not likely to be cited for a variety of subjects other than the one for which you are searching.

The sample citation search on the attached page locates papers on strong coupling theory which have cited two earlier articles in Helvetica Physica Acta and Physical Review.

STANFORD UNIVERSITY LIBRARIES

Sample search using BALLOTS/SPIRES on-line search system

COMPUTER PROMPTS APPEAR IN UPPER CASE

USER RESPONSES ARE IN LOWER CASE

```
COMMAND? spires
* WELCOME TO SPIRES
SEARCH? ipf
FIND? author martin and title computer and date after 1968
AUTHOR SEARCH FOR ...MARTIN
TITLE WORD SEARCH FOR ...COMPUTER
DATE SEARCH FOR...AFTER 1968
     1 REFERENCE(S) ACCUMULATED
? type extended
```

EXTENDED FORM OF ON-LINE COMPUTER OUTPUT

ID:	10523-2
TITLE:	Telecommunications and the computer.
AUTHOR:	Martin, James
DATE:	1969
TOTAL PRICE:	$14.25
PLACE/PUBLISHER:	Prentice-Hall
ORDER INFORMATION:	lc
BUDGET ACCOUNT NUMBER:	NTH001
SERIES STATEMENT:	Prentice-Hall series in automatic computation
TYPE OF PROCUREMENT:	po
MATERIAL RECEIPT:	7/23/69;s
INVOICE RECEIPT:	7/28/69;s;A15946893
BUDGET AMOUNT:	$14.25; $.72
MAIN ENTRY INDICATOR:	a

OPTION?

jw/sjb
9/23/69

STANFORD UNIVERSITY
Project SPIRES/BALLOTS On-Line Searching
September 26, 1969

PROMPT	RESPONSE	EXPLANATION	
SEARCH?	ipf	Library [In Process File]	FILES AVAILABLE TO SEARCH
	preprint	High Energy Physics	
	afhist	[African History]	
	geology	Geology Periodicals	
	eric	[Education Resources Information Center]	
FIND?	a	[Author]	INDEXED DATA ELEMENTS
	ca	[Corporate Author]	
	cf	[Conference Author]	
	ti	[Title]	
	d	[Date] (Subset of other index entries)	
	id	[ID Number]	
	tp	[Topic] (Not used for ipf and preprint)	
	and		LOGICAL CONNECTORS
	not		
	or		
?	(DATA ELEMENTS - see above)		
	(LOGICAL CONNECTORS - see above)		
	backup	(returns search to previous step)	
	restart	(clears present search)	
	type	(lists short form of output)	
	type extended	(lists entire copy)	
OPTION?	restart	(clears present search)	
	search	(continues present search)	
	type		
	type extended		
	exit	(exits user from SPIRES)	
	show news	(lists out news about system)	
BACKUP?	yes		
	no		
	to spires	(Legal after all prompts except BACKUP? allows user to send comments on the use of the system to the SPIRES group.)	

? serves as an implicit "and" between lines.
Search statements may be constructed on any word or words in the title, and on any form of the author which includes surnames.
Search statements may be continued beyond one line by use of the symbol @.
Words and names may be truncated after the third letter by use of the pound sign: e.g. Smi#.
Date searches must always follow another element search.
Date searches are formated: d 1969: d before 1969; d after 1968; d from 1965 thru 1968.
Two digit year representations and standard abbreviations for months are accepted.

PROMPT	RESPONSE	EXPLANATION	
SEARCH?	ipf	Library (In Process File,	FILES
	preprint	High Energy Physics	AVAILABLE
	afhist	(African History)	TO SEARCH
	geology	Geology Periodicals	
	eric	(Information Center)	
FIND?	a	(Author)	INDEXED
	ca	(Corporate Author)	DATA
	cf	(Conference Author)	ELEMENTS
	ti	(Title)	
	d	(Date) (Subset of other index entries)	
	id	(ID Number)	
	tp	(Topic) (Not used for ipf and preprint)	
	and		LOGICAL
	not		CONNECTORS
	or		
	(DATA ELEMENTS - see above)		
	(LOGICAL CONNECTORS - see above)		
	backup	(returns search to previous step)	
	restart	(clears present search)	
	type	(lists short form of output)	
	type extended	(lists entire copy)	
OPTION?	restart	(clears present search)	
	search	(continues present search)	
	type		
	type extended		
	exit	(exits user from SPIRES)	
	show news	(lists out news about system)	
BACKUP?	yes		
	no		
	to spires	(Legal after all prompts except BACKUP? allows user to send comments on the use of the system to the SPIRES group.)	

"?" serves as an implicit "and" between lines.
Search statements may be constructed on any word or words in the title, and on any form of the author which includes surnames.
Search statements may be continued beyond one line by use of the symbol @.
Words and names may be truncated after the third letter by use of the pound sign: e.g. Smi#.
Date searches must always follow another element search.
Date searches are formatted:d 1969: d before 1969; d after 1968; d from 1965 thru 1968.
Two digit year representations and standard abbreviations for months are accepted.

Sample Searching Arguments using BALLOTS/SPIRES System

```
1.  COMMAND? spires
2.  *Welcome to SPIRES
3.  SEARCH?  ipf
4.  FIND? ti intimate
5.  TITLE WORD SEARCH FOR... INTIMATE
6.          3  DOCUMENT(S) ACCUMULATED
7.  ? ti enemy
8.  TITLE WORD SEARCH FOR... ENEMY
9.          1 DOCUMENT(S) ACCUMULATED
10. ? type extended

11. ID:                     2977-2
12. AUTHOR:                 Bach, George Robert, 1914-    Wyden, Peter, Joint author.
13. TITLE:                  The intimate enemy;  how to fight fair in love and marriage
14. PLACE/PUBLISHER:        New York;  Morrow
15. DATE:                   1969

16. OPTION? restart
17. FIND? a george bach and a peter wyden and ti intimate and ti marriage and d@
18. ? after June 1968
19. AUTHOR SEARCH FOR... GEORGE BACH
20. AUTHOR SEARCH FOR... PETER WYDEN
21. TITLE WORD SEARCH FOR... INTIMATE
22. TITLE WORD SEARCH FOR... MARRIAGE
23. DATE SEARCH FOR... AFTER JULY 1, 1968
24.         1  DOCUMENT(S) ACCUMULATED
25. type extended

26. ID:                     2977-2
27. AUTHOR:                 Bach, George Robert, 1914-   Wyden, Peter, Joint author.
28. TITLE:                  The intimate enemy;  how to fight fair in love and marriage
29. PLACE/PUBLISHER:        New York;  Morrow
30. DATE:                   1969

31. OPTION? restart
32. FIND? a may
33. AUTHOR SEARCH FOR... MAY
34.         2  DOCUMENT(S) ACCUMULATED
35. ? d before 1920
36. DATE SEARCH THRU 1919
37.         0  DOCUMENT(S) ACCUMULATED
38. BACKUP? yes
39. SEARCH RESULTS RESET TO LAST        2 DOCUMENTS
40. ? d from 1919 thru 1967
41. DATE SEARCH FROM JAN-1-1919 THRU 1967
42.         1  DOCUMENT(S) ACCUMULATED
43. ? type extended

44. ID:                     4059-6
45. AUTHOR:                 May, Leopold, comp.
46. TITLE:                  Spectroscopic tricks.
47. PLACE/PUBLISHER:        New York;  Plenum Press

48. OPTION? exit
```